Tools for Visualising MARS Spectral CT Datasets

A thesis submitted in fulfilment of the
requirements for the degree of
Doctor of Philosophy
by
Alexander I. Chernoglazov

University of Canterbury
2016

Abstract

This thesis investigates the requirements and techniques for effective visualisation of MARS spectral CT datasets. It extends traditional CT data visualisation methods and develops novel tools and graphical user interfaces tailored to spectral CT data. These tools have been used to visualise and analyse a range of datasets acquired during pre-clinical research with the MARS molecular imaging system. The results demonstrate that they are capable of meeting the goals of current pre-clinical studies.

Spectral computed tomography (spectral CT) is a novel medical imaging technology that measures several energy ranges (energy bins) of the x-ray spectrum to produce multiple CT datasets. The primary advantage of spectral CT over single- and dual-energy CT is improved material discrimination, which allows for precise targeting of multiple materials of interest in a single scan.

The MARS molecular imaging system is a small specimen spectral CT scanner capable of acquiring up to eight energy bins simultaneously. Energy bin data can be processed to identify and quantify multiple materials. This thesis is the first work to study the visualisation of spectral CT material data in detail, as no prior research into the process or the tools for performing this task could be identified.

This thesis examines the use of various visualisation techniques for displaying the results of pre-clinical research with the MARS molecular imaging system. These techniques include 2D slice visualisation, direct volume rendering, magic lenses, and threshold intensity projection. Particular attention is paid to the fusion of energy and material information during visualisation, and a novel transfer function editor is developed for working with both energy, and material data comprising a MARS spectral CT dataset.

This thesis also describes a suite of tools for improving the visualisation of MARS spectral CT data. This includes tools for reducing the effects of occlusion during 3D rendering, for editing volume data during visualisation, and measurement functions tailored to spectral CT data analysis.

Deputy Vice-Chancellor's Office
Postgraduate Office

UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Co-Authorship Form

This form is to accompany the submission of any thesis that contains research reported in co-authored work that has been published, accepted for publication, or submitted for publication. A copy of this form should be included for each co-authored work that is included in the thesis. Completed forms should be included at the front (after the thesis abstract) of each copy of the thesis submitted for examination and library deposit.

Please indicate the chapter/section/pages of this thesis that are extracted from co-authored work and provide details of the publication or submission from the extract comes:

Publication:

Rajendran, K., Tredinnick, S., de Ruiter, N., **Chernoglazov, A**., Woodfield, T., Butler, A., and Anderson, N. *Metal Artefact Reduction in Ortohopaedic Implants Using MARS Spectral CT*. (2015) Accepted for presentation during the 21st Annual Scientific Meeting of the Australian & New Zealand Orthopaedic Research Society (ANZORS). October 2-4 2015, Auckland, New Zealand.

No text and no figures have been directly extracted; however, the dataset acquired, processed and visualised for this publication is described in Chapter 9, section 9.5.1, pages 262-265.

Please detail the nature and extent (%) of contribution by the candidate:

For this publication, I worked with K. Rajendran to create a 3D visualisation that clearly shows the interface between bone and a titanium screw. I have used the software developed over the course of my research to design a transfer function that separates these two materials, and produced an image included in this paper. Contribution - 15%

Certification by Co-authors:

If there is more than one co-author then a single co-author can sign on behalf of all
The undersigned certifies that:

- The above statement correctly reflects the nature and extent of the PhD candidate's contribution to this co-authored work
- In cases where the candidate was the lead author of the co-authored work he or she wrote the text

Name: *Kishore Rajendran* Signature: R. Kishore Date: *4 August 2015*

Deputy Vice-Chancellor's Office
Postgraduate Office

UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Co-Authorship Form

This form is to accompany the submission of any thesis that contains research reported in co-authored work that has been published, accepted for publication, or submitted for publication. A copy of this form should be included for each co-authored work that is included in the thesis. Completed forms should be included at the front (after the thesis abstract) of each copy of the thesis submitted for examination and library deposit.

Please indicate the chapter/section/pages of this thesis that are extracted from co-authored work and provide details of the publication or submission from the extract comes:

Publication:

R. Aamir, **A. Chernoglazov,** C. J. Bateman, A. P. H. Butler, P. H. Butler, N. G. Anderson, S. T. Bell, R. Panta, J. L. Healy, J. L. Mohr, K. Rajendran, M. F. Walsh, N. J. A. de Ruiter, S. P. Gieseg, T. Woodfield, P. F. Renaud, L. Brooke, S. Abdul-Majid, M. Clyne, R. Glendenning, P. J. Bones, M. Billinghurst, C. Bartneck, H. Mandalika, R. Grasset, N. Schleich, N. Scott, S. J. Nik, A. Opie, T. Janmale, D. N. Tang, D. Kim, R. M. Doesburg, R. Zainon, J. P. Ronaldson, N. J. Cook, D. J. Smithies, K. Hodge. (2014) *MARS spectral molecular imaging of lamb tissue: data collection and image analysis,* Journal of Instrumentation, Vol. 9, Num. 2.

No text or figures have been directly extracted from this publication; however, the dataset acquired, processed, and visualised during this study is described in Chapter 9, section 9.3.1, pages 253-255.

Please detail the nature and extent (%) of contribution by the candidate:

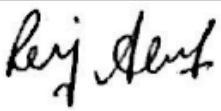
During this study, I worked with R. Aamir to visualise the distribution of lipid, calcium, and water inside a piece of lamb meat scanned with the MARS molecular imaging system. I have used the software developed over the course of my research to produce the images that clearly show the structure of this dataset and the distribution of marbling patterns inside the meat. Contribution - 15%

Certification by Co-authors:

If there is more than one co-author then a single co-author can sign on behalf of all

The undersigned certifys that:

- The above statement correctly reflects the nature and extent of the PhD candidate's contribution to this co-authored work
- In cases where the candidate was the lead author of the co-authored work he or she wrote the text

Name: *Dr Aamir Younis Raja* Signature:  Date: *04/08/2015*

Deputy Vice-Chancellor's Office
Postgraduate Office

UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Co-Authorship Form

This form is to accompany the submission of any thesis that contains research reported in co-authored work that has been published, accepted for publication, or submitted for publication. A copy of this form should be included for each co-authored work that is included in the thesis. Completed forms should be included at the front (after the thesis abstract) of each copy of the thesis submitted for examination and library deposit.

Please indicate the chapter/section/pages of this thesis that are extracted from co-authored work and provide details of the publication or submission from the extract comes:

Publication:

Panta, R. K.; Bateman, C. J.; Healy, J. L.; **Chernoglazov, A**.; Gieseg, S. P.; Butler, A. P. H.; Butler, P. H., and Anderson, N. G. Implementing Spectral Molecular Imaging (Spectral CT) in Soft Tissue. (2013) National Conference on Medical Physics (AMPICON), Kolkata, India.

No text or images were directly copied from this publication; however, the dataset acquired, processed and visualised during this study is described in Chapter 9, section 9.2.1, pages 247-250.

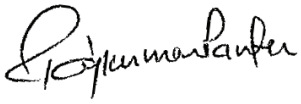
Please detail the nature and extent (%) of contribution by the candidate:

For this publication, I worked with R. Panta to visualise the distribution of lipid, water, and calcium inside an atherosclerotic plaque dataset acquired by the MARS molecular imaging system. I have used the software developed over the course of my research to display the overall structure of the plaque and the location of the calcifications inside it. Contribution - 10%

Certification by Co-authors:

If there is more than one co-author then a single co-author can sign on behalf of all
The undersigned certifys that:

- The above statement correctly reflects the nature and extent of the PhD candidate's contribution to this co-authored work
- In cases where the candidate was the lead author of the co-authored work he or she wrote the text



Name: *Raj Kumar Panta* Signature: Date: *04/08/201*

Deputy Vice-Chancellor's Office
Postgraduate Office

UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Co-Authorship Form

This form is to accompany the submission of any thesis that contains research reported in co-authored work that has been published, accepted for publication, or submitted for publication. A copy of this form should be included for each co-authored work that is included in the thesis. Completed forms should be included at the front (after the thesis abstract) of each copy of the thesis submitted for examination and library deposit.

Please indicate the chapter/section/pages of this thesis that are extracted from co-authored work and provide details of the publication or submission from the extract comes:

Publication:

K. Rajendran, M. F. Walsh, N. J. A. de Ruiter, **A. I. Chernoglazov**, R. K. Panta, A. P. H. Butler, P. H. Butler, S. T. Bell, N. G. Anderson, T. B. F. Woodfield, S. J. Tredinnick, J. L. Healy, C. J. Bateman, R. Aamir, R. M. N. Doesburg, P. F. Renaud, S. P. Gieseg, D. J. Smithies, J. L. Mohr, V. B. H. Mandalika, A. M. T. Opie, N. J. Cook, J. P. Ronaldson, S. J. Nik, A. Atharifard, M. Clyne, P. J. Bones, C. Bartneck, R. Grasset, N. Schleich, M. Billinghurst. (2014), *Reducing beam hardening effects and metal artefacts in spectral CT using Medipix3RX*, Journal of Instrumentation, Vol. 9, Num. 3.

No text and no figures have been directly extracted; however, the dataset acquired, processed and visualised for this publication is described in Chapter 9, section 9.5.3, pages 266-268.

Please detail the nature and extent (%) of contribution by the candidate:

For this publication, I worked with K. Rajendran to visualise the effects of beam hardening artefacts on MARS spectral CT data. My task was to produce images (using 3D visualisation techniques, in particular, direct volume rendering) that clearly display these artefacts and the improvement achieved by using the method proposed in this paper. Contribution - 8%

Certification by Co-authors:

If there is more than one co-author then a single co-author can sign on behalf of all

The undersigned certifies that:

- The above statement correctly reflects the nature and extent of the PhD candidate's contribution to this co-authored work
- In cases where the candidate was the lead author of the co-authored work he or she wrote the text

Name: *Kishore Rajendran* Signature:  Date: *4 August 2015*

Deputy Vice-Chancellor's Office
Postgraduate Office

UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

Co-Authorship Form

This form is to accompany the submission of any thesis that contains research reported in co-authored work that has been published, accepted for publication, or submitted for publication. A copy of this form should be included for each co-authored work that is included in the thesis. Completed forms should be included at the front (after the thesis abstract) of each copy of the thesis submitted for examination and library deposit.

Please indicate the chapter/section/pages of this thesis that are extracted from co-authored work and provide details of the publication or submission from the extract comes:

M. F. Walsh, S. J. Nik, S. Procz, M. Pichotka, S. T. Bell, C. J. Bateman, R. M. N. Doesburg, N. J. A. de Ruiter, A. I. Chernoglazov, R. K. Panta, A. P. H. Butler, P. H. Butler (2013), *Spectral CT data acquisition with Medipix 3.1,* Journal of Instrumentation, Vol. 8, Num. 10

No text and/or figures were directly extracted from this publication; however, the dataset acquired, processed, and visualisation during this study is described in Chapter 9, section 9.6, pages 269-270.

Please detail the nature and extent (%) of contribution by the candidate:

I have prepared the dataset acquired during this study for visualisation, designed a transfer function that clearly separated different tissues, and generated an image used in the paper. Contribution - 5%

Certification by Co-authors:

If there is more than one co-author then a single co-author can sign on behalf of all

The undersigned certifys that:

- The above statement correctly reflects the nature and extent of the PhD candidate's contribution to this co-authored work
- In cases where the candidate was the lead author of the co-authored work he or she wrote the text

Name: *Niels de Ruiter* Signature:  Date: *07/08/2015*

Table of Contents

List of Figures

List of Tables

Acknowledgments

Dr. Mark Billinghurst
Dr. Anthony Butler
Dr. Philip Butler
Dr. Niels de Ruiter
Dr. Raja Aamir
Dr. Christopher Bateman
Harish Mandalika
Kishore Rajendran
Raja Panta

The MARS project team
Everyone at the HITLabNZ
Family and friends

Academic Contributions

Peer-reviewed journal articles

1. M. F. Walsh, S. J. Nik, S. Procz, M. Pichotka, S. T. Bell, C. J. Bateman, R. M. N. Doesburg, N. J. A. de Ruiter, **A. I. Chernoglazov**, R. K. Panta, A. P. H. Butler, P. H. Butler (2013), Spectral CT data acquisition with Medipix 3.1, Journal of Instrumentation, Vol. 8, Num. 10, P10012.

   This paper reviewed the behaviour of the Medipix 3.1 detector and presented the results of a scan of a mouse injected with gold nanoparticles. My contribution involved producing the 3D visualisation of the region of interest where these nanoparticles have aggregated. This involved designing a complex transfer function to differentiate between soft tissue, gold, and bone.

2. R. Aamir, **A. Chernoglazov**, C. J. Bateman, A. P. H. Butler, P. H. Butler, N. G. Anderson, S. T. Bell, R. Panta, J. L. Healy, J. L. Mohr, K. Rajendran, M. F. Walsh, N. J. A. de Ruiter, S. P. Gieseg, T. Woodfield, P. F. Renaud, L. Brooke, S. Abdul-Majid, M. Clyne, R. Glendenning, P. J. Bones, M. Billinghurst, C. Bartneck, H. Mandalika, R. Grasset, N. Schleich, N. Scott, S. J. Nik, A. Opie, T. Janmale, D. N. Tang, D. Kim, R. M. Doesburg, R. Zainon, J. P. Ronaldson, N. J. Cook, D. J. Smithies, K. Hodge (2014), MARS spectral molecular imaging of lamb tissue: data collection and image analysis, Journal of Instrumentation, Vol. 9, Num. 2.

   This publication described the use of the MARS molecular imaging system for lamb soft tissue imaging. The raw, pre-processed, and reconstructed data is available at: `http://hdl.handle.net/10092/8531`.

   For this publication, I have worked with R. Aamir to present the distribution of fat-like and water-like materials in a piece of lamb meat in a clear fashion. This involved creating transfer functions that assigned distinct colours to each material and using clipping planes to show the structure inside the dataset.

3. K. Rajendran, M. F. Walsh, N. J. A. de Ruiter, **A. I. Chernoglazov**, R. K. Panta, A. P. H. Butler, P. H. Butler, S. T. Bell, N. G. Anderson, T. B. F. Woodfield, S. J. Tredinnick, J. L. Healy, C. J. Bateman, R. Aamir, R. M. N. Doesburg, P. F. Renaud, S. P. Gieseg, D. J. Smithies, J. L. Mohr, V. B. H. Mandalika, A. M. T. Opie, N. J. Cook, J. P. Ronaldson, S. J. Nik, A. Atharifard, M. Clyne, P. J. Bones, C. Bartneck, R. Grasset, N. Schleich, M. Billinghurst. (2014), Reducing beam hardening effects and metal artefacts in spectral CT using Medipix3RX, Journal of Instrumentation, Vol. 9, Num. 3.

   This is the second of two publications to present spectral CT data from the MARS molecular imaging system to the wider scientific community. The paper itself presents a validation of reducing beam hardening effects through narrow energy bins. Along side the paper, the raw, normalized, reconstructed and material decomposition datasets were uploaded for public access at ***http://hdl.handle.net/10092/8851*** *.*

   For this paper, I worked with K. Rajendran, assisting him with visualisation. This involved converting the data into the format suitable for visualisation, finding the most appropriate 3D visualisation methods, colour schemes, and camera and light positions.

Refereed conference proceedings

1. Panta, R. K.; Bateman, C. J.; Healy, J. L.; **Chernoglazov, A.**; Gieseg, S. P.; Butler, A. P. H.; Butler, P. H., and Anderson, N. G. Implementing Spectral Molecular Imaging (Spectral CT) in Soft Tissue. (2013) Presented at National Conference on Medical Physics (AMPICON), Kolkata, India.

   This presentation has been given on behalf of the MARS team by R. Panta, who has worked on assessing the composition of atherosclerotic plaques using the MARS molecular imaging system.

   Prior to the conference, I have worked with R. Panta to produce images that clearly show the distribution of materials in an atherosclerotic plaque dataset scanned by the MARS system. This dataset is described in section 9.2.

2. Rajendran, K.; Tredinnick, S.; de Ruiter, N.; **Chernoglazov, A.**; Woodfield, T.; Butler, A., and Anderson, N. Metal Artefact Reduction in Orthopaedic Im-

plants Using MARS Spectral CT. (2015) Accepted for presentation during the 21st Annual Scientific Meeting of the Australian & New Zealand Orthopaedic Research Society (ANZORS). October 2-4 2015, Auckland, New Zealand.

This presentation will be given on behalf of the MARS team by K. Rajendran, and concerns the use of the MARS system for metal artefact reduction. This work demonstrates that configuring the MARS system to use narrow energy bins leads to a significant improvement in the quality of the interface between bone and metal.

I have worked with K. Rajendran to produce 3D renderings that clearly show the improvement in the image quality when narrow energy bins are acquired by the MARS system. This is important for imaging orthopaedic implants, as the interface between bone and metal is a region that is usually significantly affected by beam hardening artefacts. The MARS system is able to greatly reduce the severity of the problem. The dataset acquired, processed, and visualised for this paper is described in section 9.5.1.

To be submitted for peer review in the near future

1. **A. I. Chernoglazov**, N. J. A. de Ruiter, V. B. H. Mandalika, C. J. Bateman, K. Rajendran, R. Aamir, S. T. Bell, N. G. Anderson, A. P. H. Butler, P. H. Butler, and M. Billinghurst. Improving Visualisation of Spectral CT Datasets. To be submitted to *Computerized Medical Imaging and Graphics* in August/September 2015.

   In this paper, I describe the algorithms for fusing energy and/or material data during 2D and 3D visualisation (sections 5.3 and 5.6), the user interface of the simple transfer function editor for material volumes (section 6.2), and the use of magic lenses to show occluded regions of interest inside spectral CT datasets (section 7.4).

2. K. Rajendran, C. Löbker, B. Schon, C. Bateman, N. de Ruiter, **A. Chernoglazov**, M. Ramyar, T. Woofield, A. Butler, and N. Anderson. Measuring articular cartilage health using multi-energy CT. To be submitted to European Radiology in August/September 2015.

   This paper describes the method for preparing and scanning an excised section

of the human knee joint with the MARS system. This study evaluates the ability of the MARS system to assess knee cartilage health. It demonstrates that the regions of healthy cartilage can be clearly differentiated from the regions of diseased cartilage because the MARS system can detect small variations in the concentration of iodine, which was used a contrast agent.

My contribution to this paper involved visualising the dataset, designing a transfer function to clearly show the diseased cartilage regions, and producing images. This work in described in detail in section 9.4.

Awards/financial grants based on a substantial assessment

- 2012-2015, Recipient of a Canterbury University PhD scholarship.

Users of my software

- Pre-clinical researchers from the MARS team, including the post-doctoral researchers and PhD students from the University of Canterbury and the University of Otago.

- Pre-clinical researchers from Lincoln University.

In addition, my software (called MARS Vision) is included in the software package distributed with the commercial version of the MARS system.

Consultation

- I am currently supporting the development of a novel hybrid user interface for spectral CT data visualisation. This interface is being implemented by Harish Mandalika (University of Canterbury PhD student), based on the software I have written over the course of my PhD research. This work is expected to result in at least two publications.

Chapter I

Introduction

This thesis describes the work conducted to develop the tools and algorithms for visualising the spectral computed tomography (CT) datasets produced by the MARS molecular imaging system. The requirements for effective visualisation are assessed, and a set of novel tools capable of analysing the structure and composition of current MARS spectral CT datasets is created. This toolset consists of several different visualisation techniques, along with a novel transfer function editor and a number of custom tools for clearly displaying regions of interest.

Spectral CT is an emerging medical imaging modality [1, 2, 3], which acquires a CT dataset for each measured energy range. Each CT dataset is captured over a certain part of the x-ray portion of the electromagnetic spectrum. In this way, the CT datasets can be considered to be different x-ray colours, just as red, green, and blue are different colours of visible light. Current CT technology includes single-energy CT (black and white), and dual-energy CT (two colours). Therefore, spectral CT can be considered to be an extension of these two imaging modalities, as shown in Figure 1.1.

Spectral CT has the potential to provide higher image quality, quantitative material discrimination and reduce the radiation dose to the patient. Pre-clinical studies have shown that spectral CT is able to differentiate between several types of tissues in the human body, as well as between the common contrast agents used in CT imaging [4, 5, 6].

Spectral CT technology has been developing rapidly over the past decade, but has not yet been introduced into clinical practice. New developments include advances in detector design and algorithms for processing spectral CT data and extracting material information. For more information, refer to Chapter 2 of this thesis, or to the overviews by Fornaro et al. [1], Anderson and Butler [2] or Taguchi et al. [3].

In addition to technical developments, a variety of clinical applications of spectral CT have been explored [7, 8, 9, 10, 11, 12]. Spectral CT data visualisation is

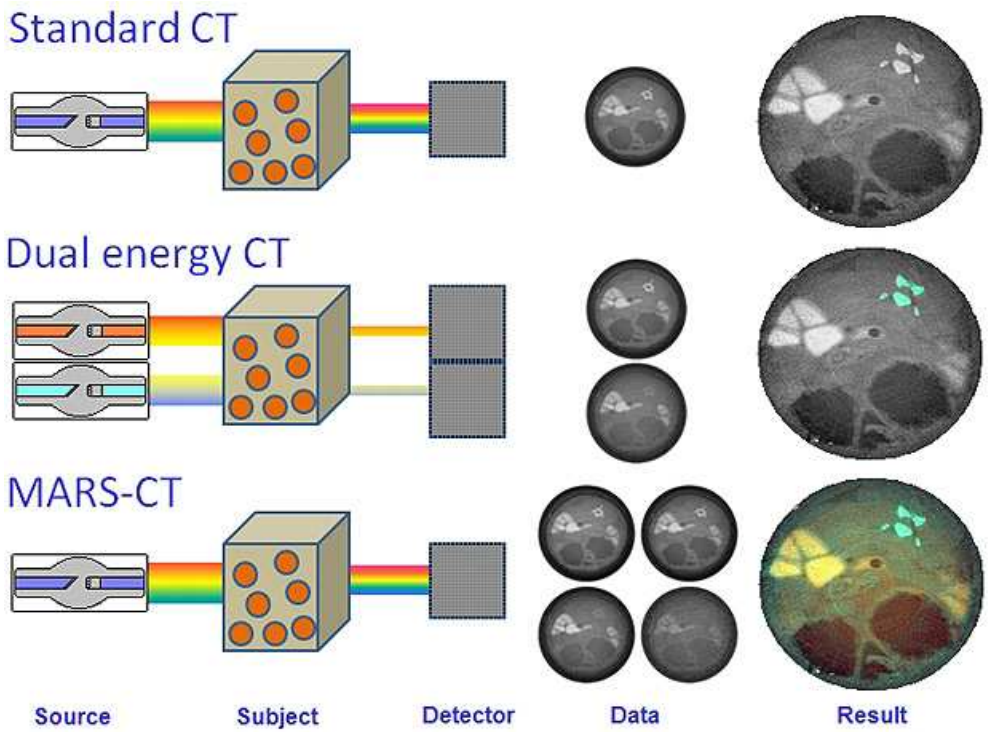


Figure 1.1: Comparison of current x-ray CT technologies. Top: single-energy CT. Middle: dual-energy CT, using two x-ray sources and two detectors. Bottom: spectral CT, using a photon-counting detector, as implemented by the MARS molecular imaging system (section 2.1.3). Image courtesy of the MARS project.

a special case of multi-variate data visualisation, where energy and material information may be visualising simultaneously. However, the process of visualising spectral CT data, and the tools and algorithms needed for carrying out this task have not been studied in detail.

This research project addresses this issue by investigating how common CT visualisation tools may be enhanced for spectral CT visualisation, and how novel tools that take advantages of the unique properties of spectral CT datasets may be developed.

After considering the characteristics of spectral CT data (section 3.2), the current image quality problems (section 3.3), and reviewing the literature (Chapter 4), a custom set of new tools was developed. This set includes:

- Direct volume rendering and 2D slice visualisation algorithms that allow the user to interactively fuse the data from multiple channels of a spectral CT dataset (sections 5.3 and 5.6).

- A custom graphical user interface (GUI) for assigning colour and opacity to channels during data fusion (Chapter 6).

- Overlay and magic lens tools for improving the visibility of regions of interest during 3D visualisation of spectral CT datasets (sections 7.3 and 7.4, respectively).

- A novel visualisation technique called multi-volume threshold intensity projection, which extends the traditional concepts of minimum and maximum intensity projection (section 7.2).

- Tools for performing measurements (such as histograms, profile of a line, or statistics inside a region of interest) and annotation (section 8.1), segmentation and interactive editing of volume data (sections 8.3, 8.4 and 8.5).

In conclusion, this research has:

- Assessed the data processing toolchain used for creating MARS spectral CT datasets. This was necessary, as visualisation is the last step of the toolchain, and all data processing affects its results in some manner.

- Explored the current problems in the field of spectral CT data visualisation, including the image artefacts contained in currently-available MARS spectral CT datasets, the lack of suitable visualisation tools, and the paucity of research into the algorithms for rendering spectral CT data.

- Investigated the spectral CT data types and determined how spectral CT datasets are different from other medical datasets. This step was required for the design of custom tools for spectral CT data visualisation.

- Studied the current process of spectral CT data visualisation and established which tools had previously been used for this purpose.

- Formulated a set of requirements for effective visualisation of currently-available MARS spectral CT datasets.

- Created the visualisation tools and rendering algorithms according to these requirements. These tools have been designed to take advantage of the unique properties of spectral CT datasets.

- Integrated these tools into a complete package for visualising spectral CT data that also contains measurement, analysis, and segmentation tools.

- Tested this system by visualising datasets acquired during the research into the clinical applications of spectral CT with the MARS molecular imaging system. This work has been performed alongside pre-clinical researchers from the MARS team, and has resulted in several publications [4, 9, 11, 13].

1.1 Thesis structure

This rest of this thesis is organised into 9 chapters, which are structured as follows:

Chapter 2 explores several topics related to this research. First, it explains the basics of x-ray imaging and x-ray computed tomography, as well as the advantages and potential clinical applications of spectral CT. Then, it describes the progress made by the MARS project, which currently produces a commercially-available pre-clinical spectral CT system. The aim of the MARS project is to eventually introduce spectral CT into clinical imaging. Next, it discusses the most common techniques for visualising medical volumetric datasets. Finally, it explains the architecture of modern GPUs and the features of GPU programming languages.

Chapter 3 explains the structure of the software toolchain that is used by the MARS project to acquire, process, and visualise spectral CT datasets. In addition, it describes the formats used to store MARS spectral CT datasets and the image artefacts contained in these datasets.

Chapter 4 conducts a review of techniques previously used for visualising spectral CT data and discusses the requirements for effective visualisation of current MARS spectral CT datasets.

Chapter 5 describes MARS Vision, a spectral CT data visualisation application that I have written over the course of my PhD research. MARS Vision implements the requirements discussed in Chapter 4. This chapter covers basic functionality such as data loading, 2D and 3D visualisation algorithms, and the acceleration and optimisation techniques necessary to achieve interactive performance.

Chapter 6 describes the process of designing transfer functions for the material datasets produced by the MARS software toolchain. It introduces a hybrid transfer function editor that is suitable for working with both energy, and material data.

Chapter 7 describes the implementation of three tools for minimising the effects of occlusion during 3D visualisation of spectral CT datasets. These tools, the threshold intensity projection, the overlay mode, and the magic lens, are designed to allow the user to clearly display occluded regions of interest inside the dataset, while retaining the context.

Chapter 8 describes the tools designed to assist the user of a spectral CT visualisation application, which include annotation, segmentation, volume editing and measurement tools.

Chapter 9 focuses on using the tools implemented in MARS Vision to visualise and analyse MARS datasets acquired during pre-clinical research into the applications of spectral CT imaging.

Chapter 10 analyses the impact of this research on the development of spectral CT visualisation techniques and software, and suggests future research directions. Spectral CT is still in the pre-clinical stage, so particular attention is paid to the upcoming transition to clinical imaging and the changes it will cause. This chapter concludes the thesis with a summary of academic contributions.

Note on usage of MARS spectral CT datasets:
The datasets used to test the algorithms and tools developed over the course of this research are described in detail in Chapter 9. However, many figures in preceding chapters use these datasets to illustrate certain concepts, or explain the use of certain tools. In these cases, a link to the section describing a particular dataset is provided. The naming convention is consistent throughout all chapters of this thesis.

Chapter II

Related work

This chapter discusses the technology behind x-ray computed tomography, the differences between conventional, dual-energy, and spectral CT, the techniques for visualising volumetric datasets, and the specifics of GPU programming.

Visualisation of spectral CT data is a task that is performed at the end of a data processing chain that starts with scanning an object and ends with the reconstructed dataset being explored by the user, or analysed by an algorithm. Therefore, visualisation is one of the last tasks a user of a spectral CT system will perform, and its outcome will be affected by a multitude of factors, such as scanner design and image processing and reconstruction algorithms.

First, section 2.1 describes the technology behind all variants of computed tomography, as well as the clinical applications of these technologies. Next, section 2.2 discusses traditional techniques for visualising volumetric datasets, making a particular emphasis on medical visualisation. Finally, principles of GPU programming that are related to this research are explained in section 2.3.

2.1 X-ray computed tomography

Tomography is a concept that refers to slicing or sectioning a dataset and is not unique to x-ray imaging. Modalities such as optical tomography [14] and PET (positron emission tomography) [15] also utilise tomographic reconstruction. In the context of this thesis, however, the term *CT* is always used to refer to x-ray computed tomography, unless otherwise specified. CT is also known as "CAT", which can stand for "computed axial tomography", "computer-assisted tomography", and "computer-aided tomography".

2.1.1 Conventional (single-energy range) CT

X-ray computed tomography is a medical imaging technology developed in the 1970s and commonly attributed to Sir Godfrey Hounsfield [16]. CT was first used

for clinical diagnosis in 1972, when it was used to detect a frontal lobe tumour [17]. Currently, CT is used for a large variety of clinical (for example, diagnosing tumours or bone fractures), scientific (biochemical and biomedical research) and industrial (non-destructive materials testing) applications. For a history of CT and its clinical applications readers are referred to the review by Kalender [18].

A CT scanner acquires a large number of *projection images* (also known as *projections* or *projection frames*) from different angles around the object being scanned. This is the common design feature shared by all CT scanners. It enables tomographic reconstruction, because the three-dimensional structure of an object can be reconstructed from a set of two-dimensional projections [19]. Other features, such as the shape of the beam, the size and position of the x-ray detector array and the pattern of gantry movement may vary [20]. Figure 2.1 shows one possible scanner design.

During a scan, x-rays are sent from the source to the detector. As the beam passes through the object, it is attenuated by different materials or tissues, which will absorb or scatter some photons. The photons that pass through the object are captured by the detector, which generates a projection image for each angle. An example of a CT projection image is shown in Figure 2.2. This is also known as a radiograph, or an x-ray image, or simply an x-ray.

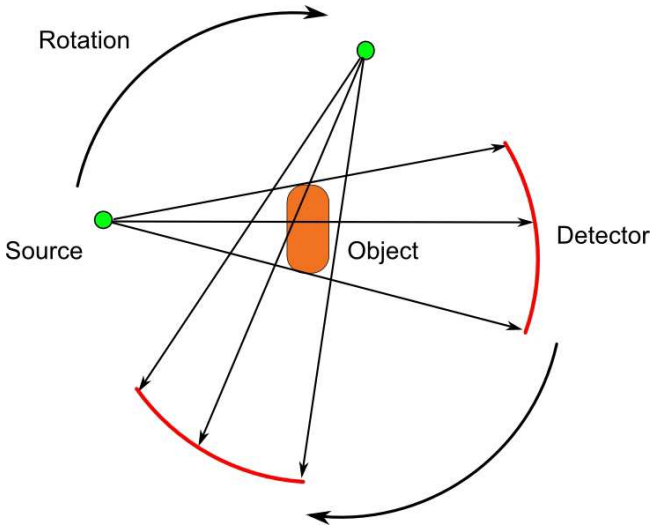


Figure 2.1: Design of a wide fan beam CT scanner. The source and the detector are fixed to the gantry, which rotates around the object being scanned and acquires projection images from multiple angles. Other configurations, where the object rotates inside a fixed gantry, are also possible.

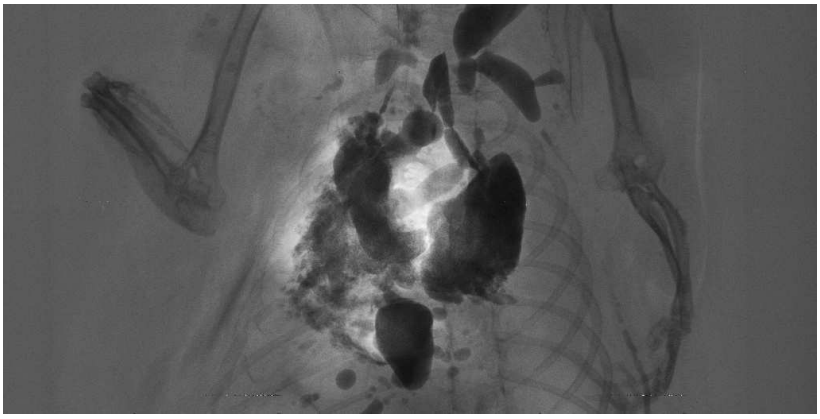


Figure 2.2: Projection image of a thoracic (chest) section of a mouse. The front feet, rib cage and some internal organs are visible. Image courtesy of the MARS project.

Projections taken over the course of a scan are reconstructed to form a 3D volumetric dataset using a variety of techniques, from filtered back projection [21] to iterative methods such as algebraic reconstruction [22]. Tomographic reconstruction techniques are beyond the scope of this thesis, but interested readers can be referred to the *Essential Physics for Medical Imaging* by Bushberg et al. [20] or to the review by Hsieh et al. [23].

Reconstructed CT datasets are usually stored as a raw binary 3D dataset, or as a set of 2D slices, as explained in section 2.2. Each element of the dataset represents the linear attenuation coefficient of photons of a certain energy range at a certain position. In clinical practice, linear attenuation coefficients are converted into Hounsfield Units (further described in section 3.2.1.1). Figure 2.3 provides an example of a reconstructed CT dataset.

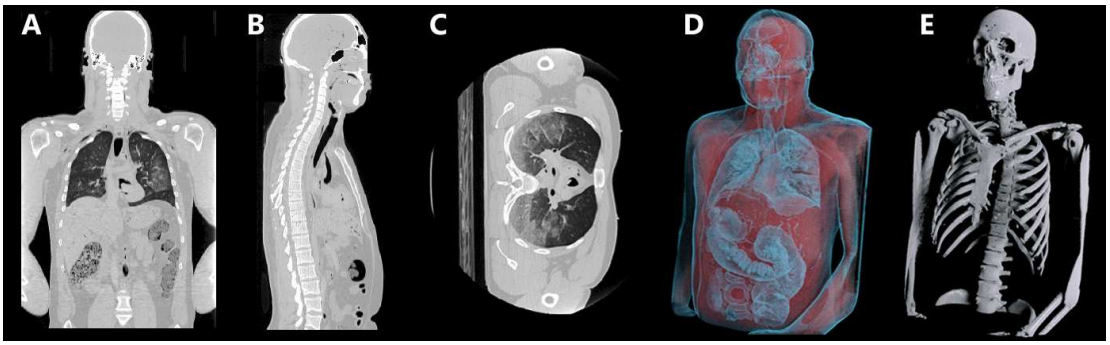


Figure 2.3: 2D slices (**A**, **B**, **C**) and 3D visualisation (**D**, **E**) of the Visible Human Male dataset (section 9.7), showing the anatomical information provided by CT.

2.1.2 Dual-energy CT and spectral CT

Single-energy CT measures a single range, while dual-energy CT measures two ranges, and spectral CT measures three or more. It must be noted that dual-energy CT is sometimes also referred to as "spectral CT". This thesis makes a distinction between these technologies, and always uses the term "dual-energy CT" to refer to systems that acquire two energy ranges, and "spectral CT" to refer to systems that acquire three or more energy ranges.

These ranges are also referred to as *energy bins*, especially when they are disjoint (non-overlapping). Usually, the human diagnostic energy range lies approximately between 12 and 120 kiloelectronvolts (keV) [24]. An electronvolt (eV) is a unit of energy approximately equal to $1.6021766 \times 10^{-19}$ joules, and is a typical unit of energy in chemical reactions.

The idea of measuring two or more energy bins has been considered for a long time. Alvarez and Macovski in 1976 [25] and Di Chiro et al. in 1979 [26] have investigated the theoretical background of multi-energy CT imaging and demonstrated its advantages, such as higher contrast-to-noise ratio and improved material discrimination. Experiments conducted in 1986 have confirmed the viability of dual-energy CT for clinical imaging [27]. However, this technology has only entered routine clinical use around 10 years ago [28], although similar dual-energy techniques have been used by other medical imaging modalities such as radiography [29] and bone densitometry [30, 31].

The motivation for developing the technology for measuring two or more energy bins is based on the fact that the attenuation of photons through a material varies non-linearly with respect to energy [32]. This is shown in Figure 2.4, which plots the mass attenuation profiles of several materials present inside the human body (water, calcium) or used as contrast agents (barium, iodine, gadolinium). Given sufficiently precise measurements of the x-ray spectrum, these materials can be separated and quantified. Therefore, the molecular composition of organs or tissues composed of these materials can be assessed.

The following example can be used to illustrate this principle. Contrast agents such as iodine and barium are often used in clinical CT imaging to improve the visibility of certain tissues or organs of interest and other surrounding tissues. However, as shown in Figure 2.4, iodine and barium attenuate photons very similarly over most of the diagnostic energy range. In most cases, conventional CT can-

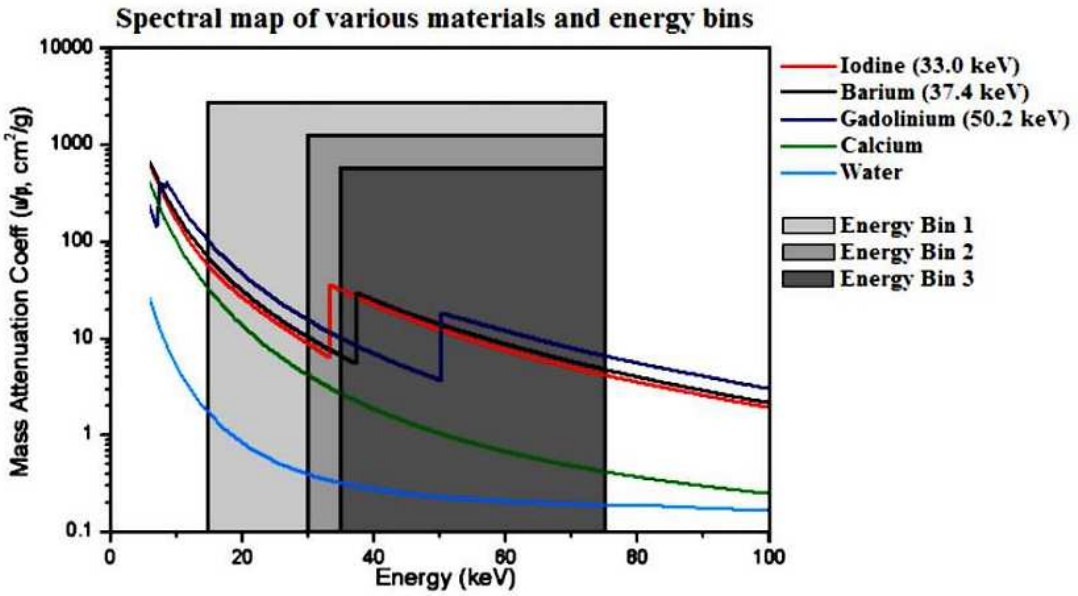


Figure 2.4: Mass attenuation and k-edges of five materials over the diagnostic range of the x-ray spectrum. Attenuation varies based on photon energy. K-edges for iodine, barium and gadolinium are sharp increases in the mass attenuation profiles of these elements. In contrast, water and calcium have no detectable k-edges in this part of the spectrum. Image provided by the MARS project.

not discriminate between these two elements (and, consequently, between contrast agents based on these elements) because they appear to have the same radiodensity [6].

The only significant difference in the mass attenuation profiles of barium and iodine is near the *k-edges*, where the photoelectric effect causes an electron from the K shell to interact with photons of a particular energy. The photon is absorbed, which increases the attenuation coefficient. Figure 2.4 shows that the k-edge for iodine is located at 33.0 keV and the k-edge for barium is located at 37.4 keV.

A technique called k-edge imaging [33] can be used to distinguish between these elements: energy bin 2 measures the total attenuation over the 30-75 keV range and includes iodine's k-edge, but the energy bin 3 (35-75 keV) does not. If these bins are acquired during a scan, the increase in attenuation caused by the k-edge of iodine will be measured in bin 2, but not in bin 3. In contrast, conventional CT will only acquire a single energy bin (for example, energy bin 1 in Figure 2.4), which will measure the entire spectrum from 15 to 75 keV. This will include the k-edges of both elements and therefore will not contain enough information to

discriminate between them.

Other methods, such as PCA (principal component analysis, a statistical technique for extracting sets of linearly uncorrelated variables from multi-variate datasets [34]) [6], or basis material decomposition [35,36], can also be used. These methods extract material information from multiple energy bins, which can be acquired by dual-energy or spectral CT systems.

Energy bins can be acquired sequentially or simultaneously. For example, Kalender et al. [27] have used a single-energy CT system, but varied the voltage of the x-ray tube to produce two different energy spectra, which were measured separately. This increased the scanning time and left the system susceptible to motion artefacts. On the other hand, photon-counting detectors used in spectral CT, such as the Medipix3 [37] used in the MARS system (section 2.1.3), acquire all energy bins simultaneously.

Simultaneous acquisition of energy bins by a single detector removes the need for registration (scaling and alignment of datasets), which is a common problem when images are acquired at different times or by different scanner hardware [38]. For example, multiple modalities, such as ultrasound and CT [39], CT and SPECT (Single positron emission computed tomography) [40], or MRI (magnetic resonance imaging) and CT datasets [41], must be registered if the diagnosis requires visualisation with multi-variate fusion.

2.1.2.1 Applications of dual-energy and spectral CT

Dual-energy CT has been shown to improve diagnosis of conditions such as gout [42, 43], endoleaks [44], cancer [45, 46] and renal stones [47]. In all of these cases, dual-energy CT is used to target specific materials, such as urate, calcium, or iodine-based contrast agents. Two energy bins acquired by dual-energy systems allow for some degree of material discrimination. However, spectral CT measures the x-ray spectrum more precisely, enhances the accuracy of material decomposition and increases the number of materials that can be targeted by a single scan.

Spectral CT is not currently used in clinical practice, and the full range of applications has not yet been established. However, it is thought that spectral CT will improve diagnosis of conditions such as fatty liver disease, osteoarthritis, atherosclerosis and cancer, and reduce the radiation dose to the patient [1,2,3]. In addition, the dose to the patient and the scanning time will likely be reduced [2].

Several prototype spectral CT scanners have been constructed [48, 49, 50, 51], and the ability of spectral CT imaging to discriminate between different materials has been confirmed by numerous studies. For example, calcium [4,5,6] gadolinium [7, 52], gold [5, 10], barium [6, 53], iodine [5, 6, 52, 54] and specific soft tissues [3, 4, 52, 55] have been identified.

Most studies have not yet reported reliable and accurate quantification of these materials, but this is expected to be achieved soon by improving existing detector technology and data processing algorithms. Accurate quantification of these materials would be useful for clinical imaging because many existing contrast agents are based on barium, iodine and gadolinium, while gold can be used in nanoparticles that deliver drugs to specific tissues such as tumours [56], and fat and calcium are naturally present in human body tissues.

Quantifying these materials allows:

1. Measuring the presence of a biomarker, which could signify the progression of a disease.

2. Measuring the drug dose needed to target biomarkers.

3. Measuring the exact volume of a target object.

In conclusion, dual-energy CT is already a widely-used medical imaging modality. Pre-clinical research using spectral CT has shown that it further improves material discrimination and quantification and increases the range of materials that can be targeted. Spectral CT is expected to grow rapidly in the near future [2] and human trials may begin as soon as 3-5 years from now [3]. Currently, it is still an expensive (largely due to the cost of the detectors) and experimental imaging technology. As of 2015, no human-sized spectral CT system is sold commercially, and no common CT extensions, such as cardiac gating, are available.

2.1.3 The MARS molecular imaging system

The MARS project is a collaboration between the University of Canterbury and a number of domestic and international partners such as the University of Otago and CERN (Conseil Européen pour la Recherche Nucléaire, the European Organisation for Nuclear Research). The goals of the MARS project are research into the clinical

Figure 2.5: The MARS scanner. Image courtesy of the MARS project.

and scientific applications of spectral CT and commercialisation of spectral CT technology.

The MARS molecular imaging system is a prototype small specimen spectral CT scanner built for small animal and sample imaging. Figure 2.5 shows a recent version of the MARS scanner. The current iteration is able to scan objects up to around 100 mm in diameter and 280 mm in length [57]. The x-ray source and detectors are mounted on the opposite sides of a gantry, which is capable of 360° rotation around the sample. The sample is fixed to an independent mount and can be be translated (moved forward and backward), which is necessary when scanning larger samples. The schematic of a recent version of the MARS scanner is shown in Figure 2.6.

The MARS scanner can use several types of detector chips that belong to the Medipix line of photon-counting detectors. Currently, variants of the Medipix3 chip, which consists of a matrix of 256×256 55×55 μm pixels [37], are used.

The energy of an incoming photon is measuring using a process called pulse height analysis. A photon creates a charge cloud when it interacts with a semicon-

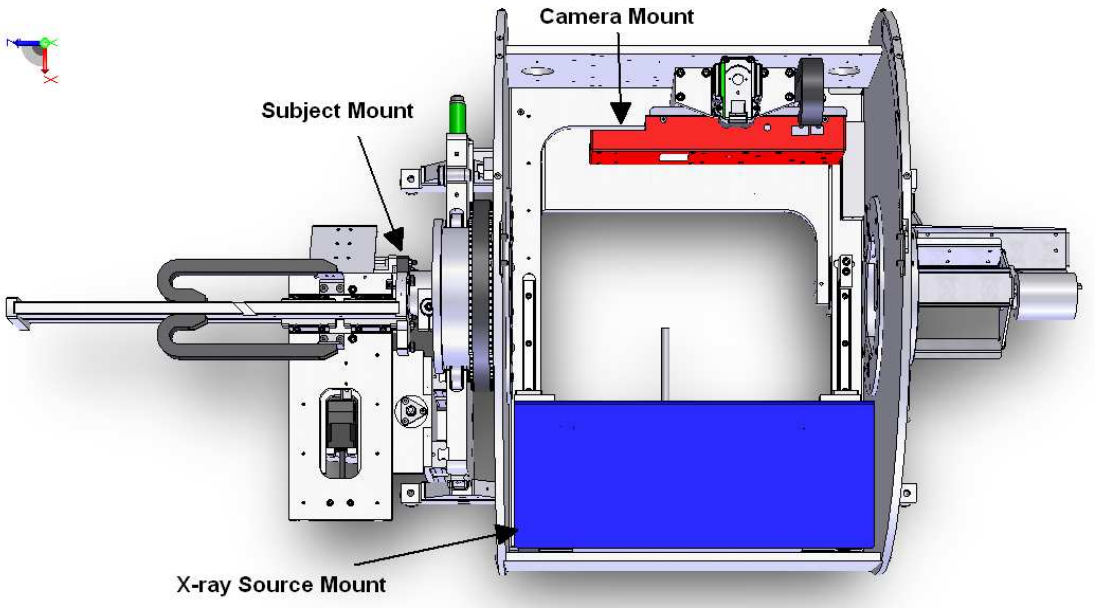


Figure 2.6: Schematic of the 2013 version of the MARS scanner, showing the camera, x-ray source and subject mount. Image courtesy of the MARS project.

ductor sensor layer, which is bonded to the ASIC (application-specific integrated circuit). Each pixel of the detector uses an amplifier, a shaper, and a discriminator to measure the height of the input pulse generated by the charge cloud. Once the energy is calculated, the photon is counted in the appropriate energy bin.

The MARS system can be configured to partition the diagnostic energy range into several energy bins, for example 15-80 keV, 23-80 keV and 35-80 keV. The lower limit is a threshold set on the detector, and the upper limit is the highest energy of the x-ray photons produced by the x-ray tube. During a scan, it acquires a set of projection images at every camera angle. Each projection measures the attenuation of photons that belong to a particular energy bin.

The MARS version 3 (v3) scanner took around 2.5 seconds to acquire a projection a single camera angle, which means that a 360° rotation with projections taken at every 1° took around 15 minutes [58]. However, the scanner is constantly being improved to reduce the total imaging time. When operated in fast mode, the MARS scanner is capable of scanning live animals, and in October 2011 the first successful scan of a sedated live mouse was performed. The current version of the scanner (v5) acquires a single projection in approximately 0.5 seconds, which reduces the scan time to around 3 minutes.

Tomographic reconstruction of projection data using a custom iterative algebraic reconstruction (ART) algorithm [22,59] results in several volumetric datasets. Following the examples above, there will be one reconstructed dataset that represents the attenuation of photons in the 15-80 keV range, another dataset for the 23-80 keV range, and so on. Narrow, non-overlapping energy bins (for example, 20-30 keV, 30-40 keV, and so on) may also be created by subtracting the photon counts in the raw projection data. Reconstructed energy bins will be referred to as *energy volumes* throughout this thesis.

Acquisition, processing and reconstruction of data is performed by a set of programs called the data processing toolchain, which is described in Chapter 3. Datasets acquired by the MARS scanner and processed by the MARS toolchain have been described by Rajendran et al. [9] and by Aamir et al. [4].

2.1.4 Summary

Computed tomography is an imaging technology that measures the attenuation of x-rays as they pass through an object and reconstructs its internal structure from a series of projections taken from different positions. CT is used for numerous clinical, scientific and industrial purposes.

In clinical practice, CT measures a single energy range and provides excellent information about the structure of the body (anatomical information). However, it is not suitable for determining the molecular compositions of tissues. Dual-energy CT measures two energy bins and provides additional material information, while spectral CT divides the spectrum into three or more energy bins and further enhances material discrimination.

Spectral CT has not yet been introduced into clinical practice. Several prototypes have been constructed and pre-clinical research has shown that this technology can improve diagnosis of conditions such as lung infections and degenerative joint disease and detect biological markers of inflammatory diseases and cancer.

2.2 Visualisation of volumetric datasets

This section discusses the techniques for visualising volumetric datasets (also referred to as *volumes*), with a particular emphasis on visualisation of medical datasets. This section explains the basic techniques, such as 2D slice visualisation, and mesh and volume rendering, that can be applied to display all types

of volumetric datasets. Specific work on spectral CT visualisation is covered in Chapter 4.

Datasets generated by tomographic reconstruction (for example, CT or MRI datasets) are usually represented as 3D arrays of volume elements, or *voxels*. Voxels are typically positioned on a regular grid in 3D space.

Volumetric datasets are most commonly stored as a set of *slices*: cross-sections of an object. Each cross-section is stored as a separate image file. Slices can be visualised as a sequence of 2D images, or integrally, as a single 3D dataset. This is done using techniques such as direct volume rendering (DVR) [60], shear warping [61], splatting [62], or mesh extraction and rendering. Display of 2D slices generally poses no problems for modern computing hardware, while 3D visualisation is much more computationally expensive.

2.2.1 2D visualisation

2D visualisation has a long history of use in medical practice, for example as part of the standard x-ray radiography visualisation process, which involves the radiologist manipulating sheets of x-ray film [64]. Currently, most medical visualisation is performed with the help of digital workstations, but 2D techniques still play an

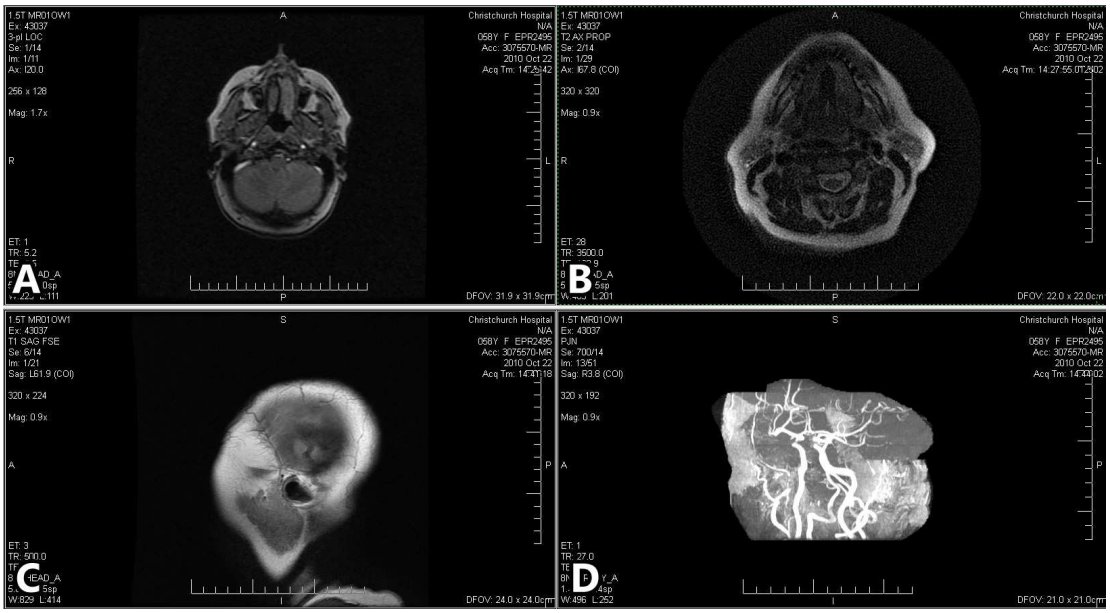


Figure 2.7: Visualisation of an MRI dataset with eFilmLite [63]. Different 2D views (**A**, **B** and **C**) are shown along with a basic 3D rendering (*D*).

important role.

The most basic 2D technique is visualising a sequence of slices by scrolling through it. This functionality is supported by specialised DICOM (Digital Imaging and Communications in Medicine [65]) viewers such as eFilmLite (Figure 2.7), or InteleViewer [66], and by general scientific or medical visualisation applications such as ImageJ [67], 3D Slicer [68], or MITK [69].

A slice is usually displayed using a greyscale colour scheme, where darker shades correspond to pixels of low attenuation, density or concentration (depending on the dataset type), and lighter shades correspond to highly attenuating areas, or areas of high concentration of a material. The appearance is normally controlled by adjusting window and level settings. The level controls the brightness, while the window controls the contrast.

This is illustrated in Figure 2.8. All data values lower than $(level - \frac{window}{2})$ (the second graph node) are coloured in black. All data values higher than $(level + \frac{window}{2})$ (the third graph node) are coloured in white. The colour for data values between these two points is interpolated between black and white, creating a greyscale gradient. Therefore, narrow windows provide good contrast over a small data range (left graph), while wide windows reduce the visible contrast, but show a larger data range (right graph).

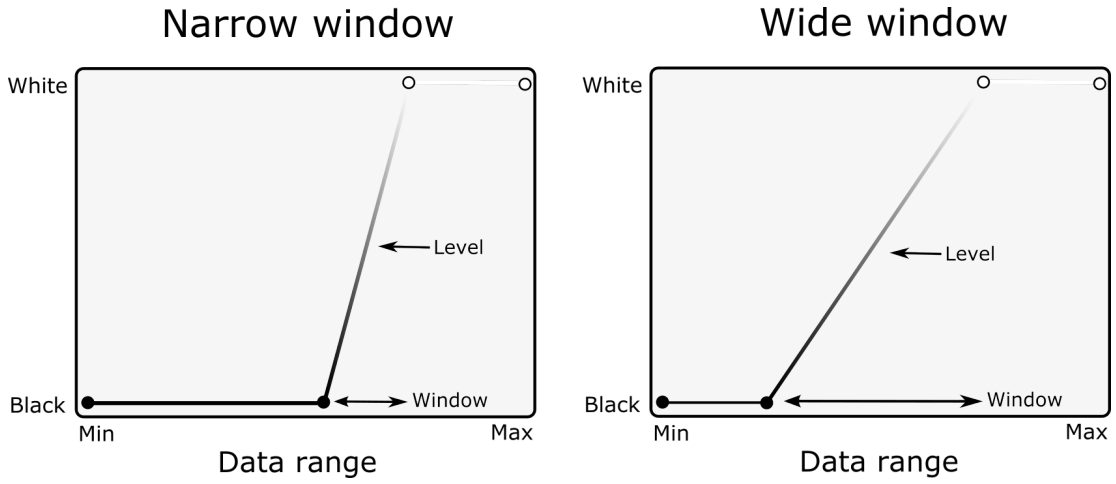


Figure 2.8: The use of window and level settings to set a greyscale colour scheme.

The reason why window and level adjustment is needed is that the data range is often too wide to be displayed using a conventional PC monitor. This means that the number of distinct shades of grey is limited (most commonly to 256) by the

display hardware. In comparison, the standard Hounsfield scale used in clinical CT imaging (section 3.2.1.1) typically ranges from -1000, which corresponds to air, to around 3000, which corresponds to dense bone. Therefore, the full dynamic range of a conventional CT dataset cannot be displayed on most monitors, making some form of image adjustment necessary.

This is illustrated in Figure 2.9. A wide window (A) allows the observer to see all tissues present in this slice, but detail is lost in some parts (for example, small variations in the radiodensity of soft tissues are invisible). A narrow window (B) can clearly show certain regions or tissues of interest, but will also hide other tissues.

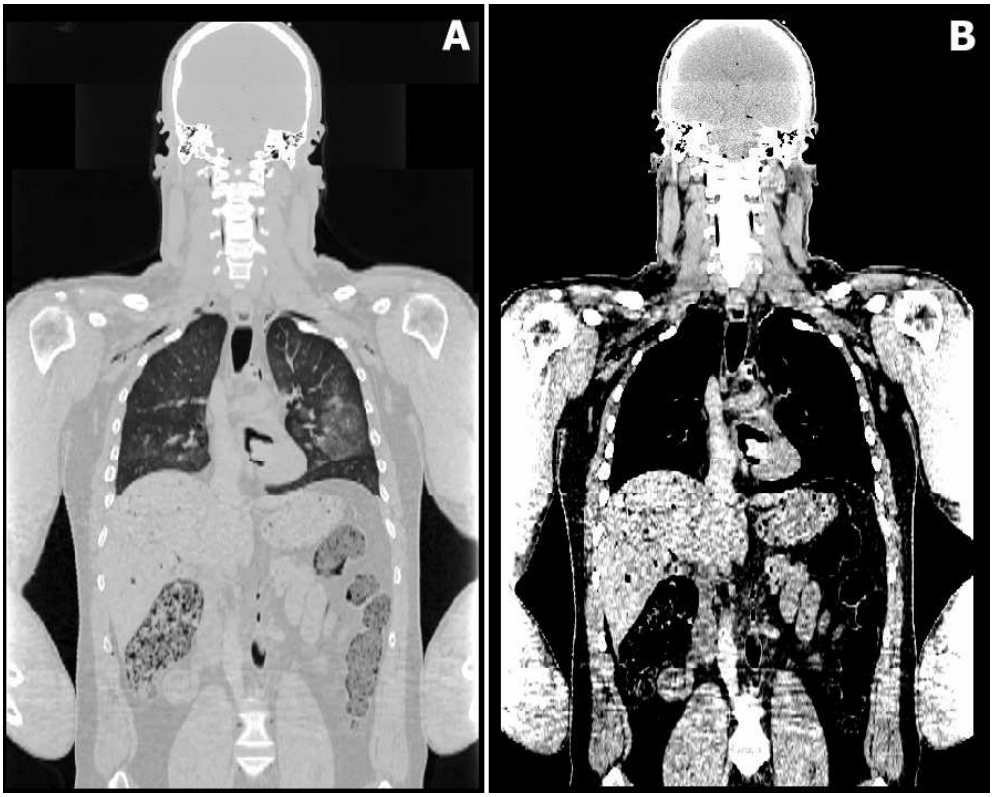


Figure 2.9: Visualisation of a slice of the Visible Human Male dataset (section 9.7.1). **A**: wide window: level = -1024, window = 3186. **B**: narrow window. Level = 0, window = 140. Window and level values are given in Hounsfield Units.

2D visualisation possesses numerous advantages, such as:

- Low hardware requirements - displaying a single slice of a volumetric dataset is a trivial computational problem, while 3D visualisation usually requires

powerful GPU hardware (section 2.3). A GPU may not be able to store the entire dataset at once if it is extremely large, like some micro-CT datasets (for example, 4096^3 voxels [70]), or if it is multi-variate, like spectral CT datasets.

Current GPUs contain limited and non-expandable amounts of video memory (also known as video random access memory, or VRAM). At the time of writing, the largest amount of VRAM available on a workstation GPU is 12 GB [71], and consumer-grade GPUs commonly contain 3-4 GB [72, 73]. Currently, a large spectral CT dataset can exceed 8 GB in size (1024^3 voxels, 16-bits per voxel, 4 energy bins [74]), which means that it must be broken up into subsets, downsampled, or compressed in order to be visualised on most GPUs. This will lower the image quality or the rendering speed.

- Greater fidelity, because a 2D slice is directly mapped to an image on the screen. In contrast, each pixel of an image produced by a volume visualisation technique usually contains data from multiple voxels that have blended together. In addition, 3D visualisation often requires certain data ranges to be hidden to minimise occlusion (as discussed in section 2.2.2). Incorrectly setting the visible and invisible data ranges can lead to features of interest being hidden.

  Figures 2.10 and 2.11 can be used to illustrate this limitation of 3D visualisation. The 3D renderings have to omit some information: in Figure 2.10, the inside of the brain not shown, while in Figure 2.11, the internal anatomy of the carp is largely hidden. Visualisation of individual 2D slices avoids this problem, as all information present in a slice can be displayed without modification or occlusion.

- Ease of interaction - standard 2D pointing tools, such as a mouse or a trackball, can be easily and naturally mapped to a pointer on the screen. This allows users to place annotations and perform measurements.

- Suitability for mobile devices such as smartphones and tablets. These devices contain CPUs and GPUs that are substantially slower than those in a typical desktop PC. While visualisation of medical datasets on mobile platforms is

not widespread, recent research has shown that it can be an effective tool in emergency medicine [75, 76].

Figure 2.10: Left: a 3D rendering of the MRI Head dataset [77]. Right: examples of 2D slices from the MRI Head dataset.

In addition, 2D visualisation is the primary technique used by radiologists [78, 79, 80, 81], who are expected to comprise the majority of future users of spectral CT [2]. 3D visualisation is commonly used for planning and carrying out complex surgical procedures (preoperative planning [82, 83, 84] and computer-assisted surgery [85, 86, 87]). However, at this time, such tasks are not carried by members of the MARS project, as most users are pre-clinical researchers working with small animals or excised tissue samples. Therefore, 2D visualisation remains an essential requirement for current MARS users, although 3D visualisation is likely to become more prevalent in the future.

2.2.2 3D visualisation of volumetric datasets

3D visualisation, or 3D rendering, refers to the set of techniques that draw a volumetric dataset as a single 3D array of voxels, as opposed to a set of independent slices (2D arrays of pixels) [88]. It must be noted that the term "reconstruction" is sometimes also used to refer to the 3D rendering techniques described in this section. This thesis uses "reconstruction" to refer to the process of tomographic

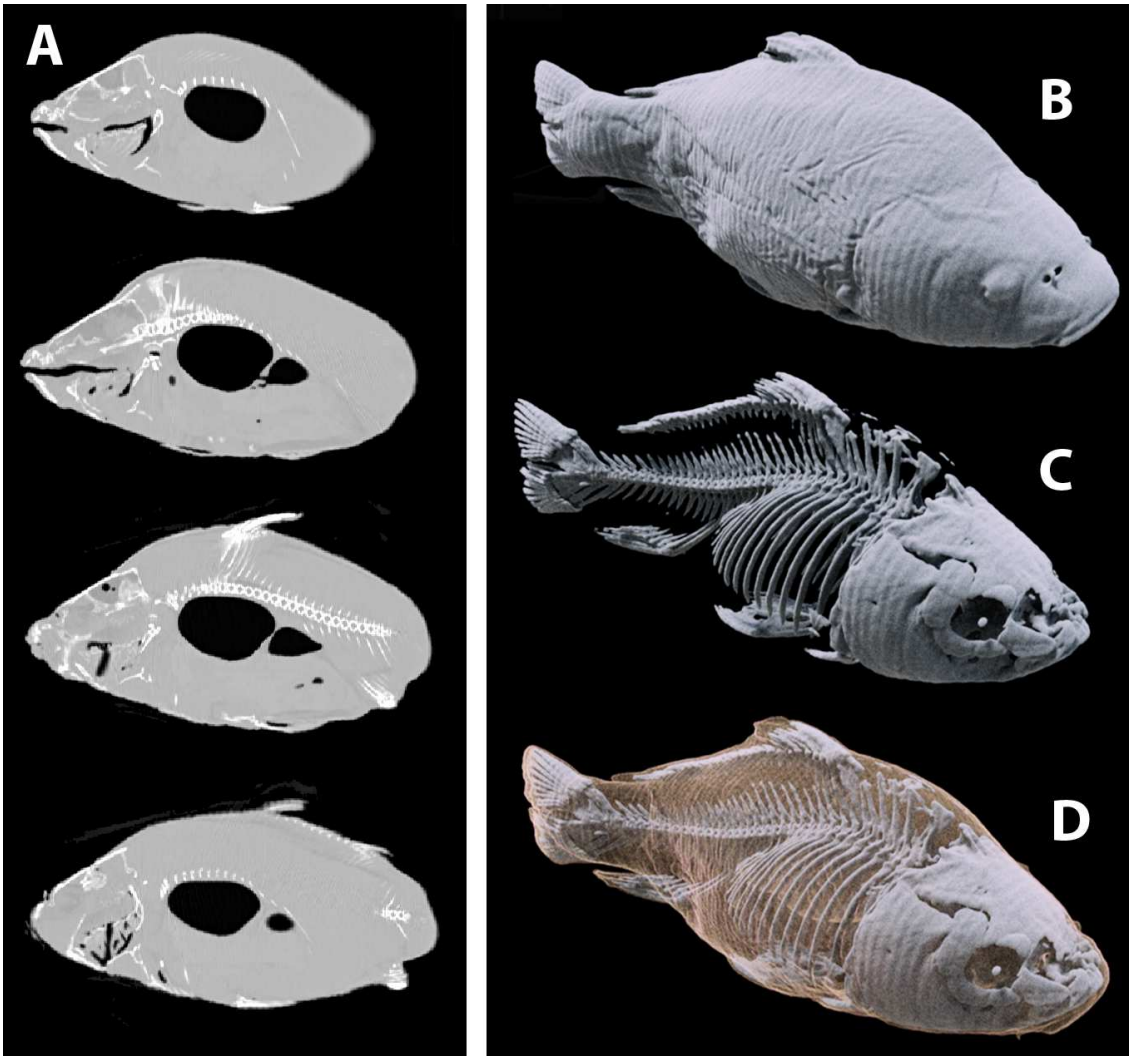


Figure 2.11: Visualisation of the Carp dataset. Left side (A): examples of slices. Right side: 3D visualisation using DVR. B: transfer function set to show skin as an opaque surface. C: transfer function set to show bones as an opaque surface. D: transfer function set to show skin as a translucent surface and bones as an opaque surface.

reconstruction (that is, the process that creates a volumetric dataset from projection data) and "3D visualisation" to refer to the techniques used for rendering this dataset on the screen to generate an image.

This approach can convey significantly more information than traditional 2D slice visualisation because regions of interest (ROIs), such as the organs in the human body, normally span across multiple slices of a CT dataset. Therefore, when viewing slices, the user must mentally "reconstruct" the structure of the object by viewing a sequence of slices. 3D rendering accomplishes the same task

by using computer hardware to visualise structures. A comparison between 2D and 3D visualisation is shown in Figures 2.10 and 2.11.

2.2.2.1 Direct volume rendering

Volume rendering is a generic term for a set of techniques for visually extracting information from volumetric datasets. Volume rendering includes both image-order, and object-order techniques. Image-order techniques, such as ray casting [89], start traversing the volume from the image plane. Object-order techniques, such as shear-warping [61] and splatting [62], start traversing the volume from some point inside the volumetric dataset. Refer to the Visualization Handbook [90] for a detailed discussion of volume rendering techniques.

The current standard is direct volume rendering (DVR) [60], which uses ray casting for traversing the volume and generating an image. This technique generates high-quality images, and various DVR algorithms can often be efficiently implemented on modern hardware (usually GPUs, as discussed in section 2.3).

DVR treats the volume of space occupied by the dataset as a cloud of gas [60]. The appearance of this cloud is calculated according to a physically plausible optical model. The choice of the optical model often determines the visual quality and the performance of a DVR algorithm [91]. This section briefly describes the emission-absorption model, which is the most commonly-used optical model used in DVR.

In DVR, a rendering pass is started by placing a virtual camera at an arbitrary position in 3D space and casting one ray from each pixel of the camera's "screen" (the *image plane*), as shown in Figure 2.12. Rays traverse the volume and accumulate colour and opacity by sampling volume data at regular (or, less-commonly, variable) intervals, as shown in Figure 2.13. The interval between samples affects image quality and rendering speed. A smaller interval (more samples per ray) results in better visual quality at the expense of performance. A larger interval between samples (fewer samples per ray) leads to the appearance of visually perceptible artefacts, but increases the rendering speed.

2.2.2.2 Interpolation

Voxels are discrete data elements on a regular grid in 3D space. However, rays may sample at any arbitrary location inside the volume, so some form of interpolation

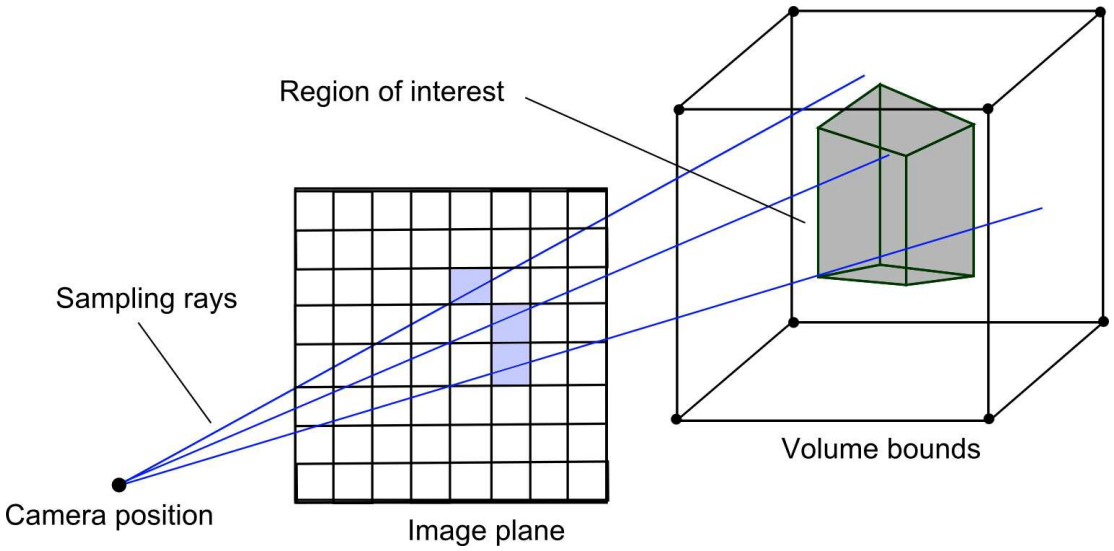


Figure 2.12: The process of volume raycasting. The camera sends rays through the image plane; rays sample inside the volume and generate an image pixel-by-pixel.

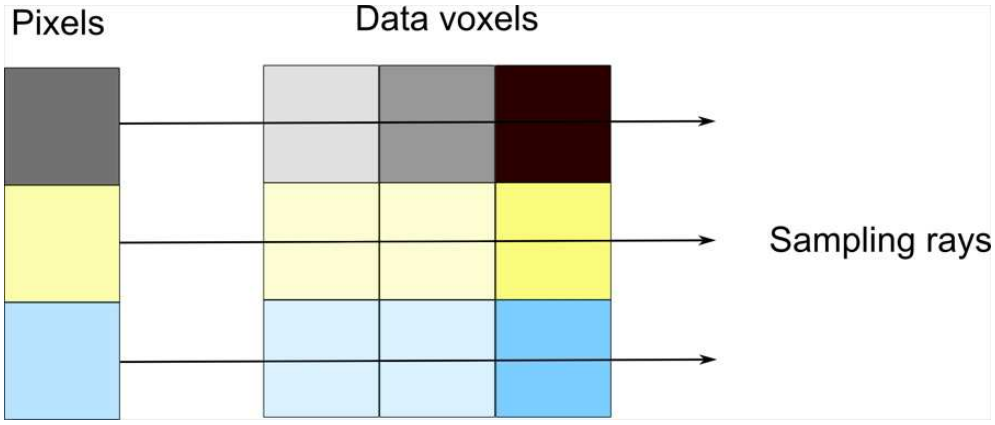


Figure 2.13: Sampling and colour accumulation during a single pass of a ray marching DVR algorithm. The volume is sampled at regular intervals along each ray, and the value at each sampling point is added to the sum over the entire ray. When the ray terminates, the colour it has accumulated is displayed as an image pixel (left column).

is usually required, as most samples will be taken at points located between several voxels.

Nearest neighbour [92] is the fastest possible interpolation technique, because it simply returns the value of the voxel closest to the sampling point. This gives the final image an unrealistic, "blocky" appearance (Figure 2.14A). For this reason, nearest neighbour is not commonly used for volume data interpolation.

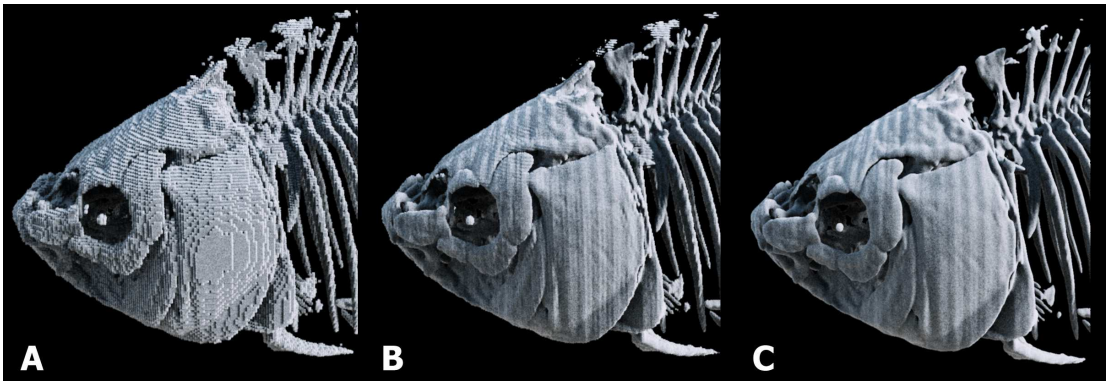


Figure 2.14: Comparison of interpolation techniques. **A**: nearest neighbour. **B**: trilinear. **C**: cubic B-spline.

Trilinear interpolation [93] is an extension of linear interpolation (interpolation along a line between two points) into three dimensions, as shown in Figure 2.15. It provides an acceptable trade-off between visual quality and performance for most purposes. An example is provided in Figure 2.14B. This method is already implemented in hardware by modern GPUs, which contain texture interpolation units and special texture caches. These units are generally used for bilinear interpolation of 2D textures, but also support trilinear interpolation of 3D textures, which is the format most commonly used to store volumetric datasets on GPUs.

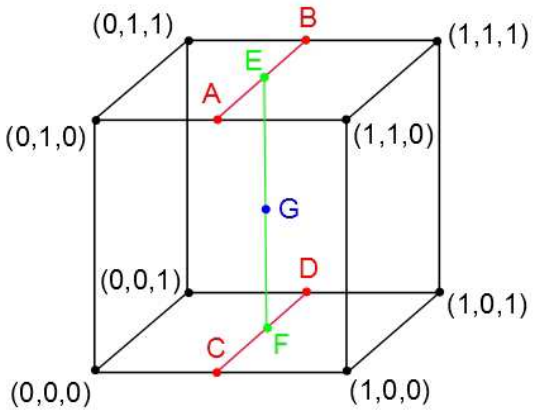


Figure 2.15: Trilinear interpolation. To interpolate a value at location G, 8 nearest voxel values are retrieved and linear interpolation is used to obtain values at A, B, C and D. Then, values at E and F are found by linear interpolation. Finally, G is calculated by linear interpolation of values at E and F.

More advanced techniques, such as cubic B-spline interpolation (also known as tricubic interpolation [94]), take into account a larger number of neighbouring voxels, assigning a different weight to each voxel based on its proximity to the sampling point [95]. This results in smoother surfaces and fewer artefacts, as shown in Figure 2.14C. However, the difference in quality, compared to trilinear interpolation, is usually not as substantial as the difference between nearest neighbour and trilinear interpolation.

For example, note that in Figure 2.14, the severity of visible artefacts is greatly reduced by using trilinear interpolation (B). Tricubic interpolation (C) eliminates more artefacts, but in practice, it is often impractical because of the reduction in rendering speed. The implementation of these interpolation techniques and their effects on the performance of a DVR algorithm for spectral CT data visualisation are further discussed in section 5.5.4.

In conclusion, the algorithm used for interpolation between voxels affects both the visual quality and performance of a visualisation algorithm [96]. Currently, trilinear interpolation is the most popular method [97].

2.2.2.3 Transfer functions, visibility of features and occlusion

Visibility of dataset features is one of the most important problems in 3D visualisation in general, regardless of the algorithm. In DVR, the physical properties of different data ranges are set through the use of a *transfer function*, which maps scalar voxel values to colour (usually in the RGB colour space) and opacity (alpha) [98] values. This means that, during rendering, a sample taken inside the dataset is mapped to a particular, but not necessarily unique, RGBA (red, green, blue, alpha) value. This is necessary, as some optical properties must be assigned in order to display the volume data. This is done for two reasons:

1. Most medical imaging modalities (including CT) do not acquire data using visible light. Therefore, elements of these datasets have no inherent colour, yet some colour must be assigned for the purposes of visualisation.

2. Features of a dataset can be emphasised or hidden by assigning colour and opacity. Therefore, a well-designed transfer function can help users analyse the dataset by classifying various materials or tissues.

In most cases, transfer functions need to be designed manually, through trial and error. It is often a complex and time-consuming process [99, 100]. Transfer function design is usually a serious problem for CT data visualisation because different materials or organs may attenuate x-rays in an identical, or nearly-identical manner. For example, consider Figure 2.16: the two hollow tubes at the front of the image are the sample itself - an excised piece of the human carotid artery. However, the wall-like structure at the back of the image (highlighted with an arrow) is a plastic tube that was used for holding the sample during scanning. The measured attenuation of the soft tissues in the artery and the plastic in the tube is identical, which makes it impossible to construct a transfer function for this dataset that displays one material, but not the other.

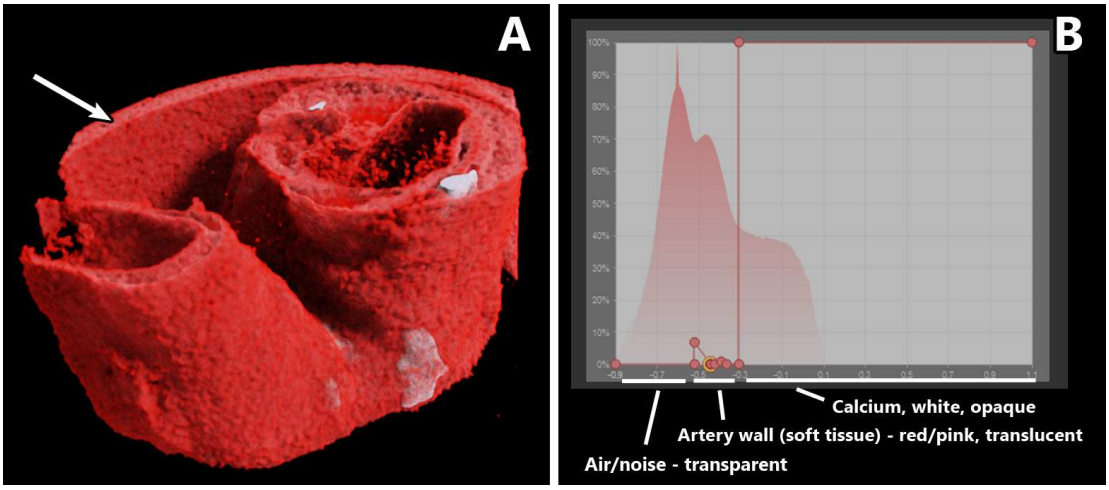


Figure 2.16: **A**: direct volume rendering of an energy volume of the Plaque72 dataset (section 9.2.1). **B**: the transfer function used to achieve this appearance.

The difficulty of transfer function design depends on the dataset, and the tissues that need to be classified. For example, Figure 2.17 shows an MRI dataset where separation of brain tissue from skull bone is a trivial task. Bone and soft tissue can also be separated during visualisation of CT datasets, as the differences in attenuation are substantial. In other cases (for example, the separation of soft and fat tissues), the differences in attenuation are much smaller. This problem, as well as the design of editors used to create transfer functions, is further explored in Chapter 6.

Occlusion of features is a problem that may be caused or solved by transfer function design, as shown in Figure 2.11. Internal structures such as bones are

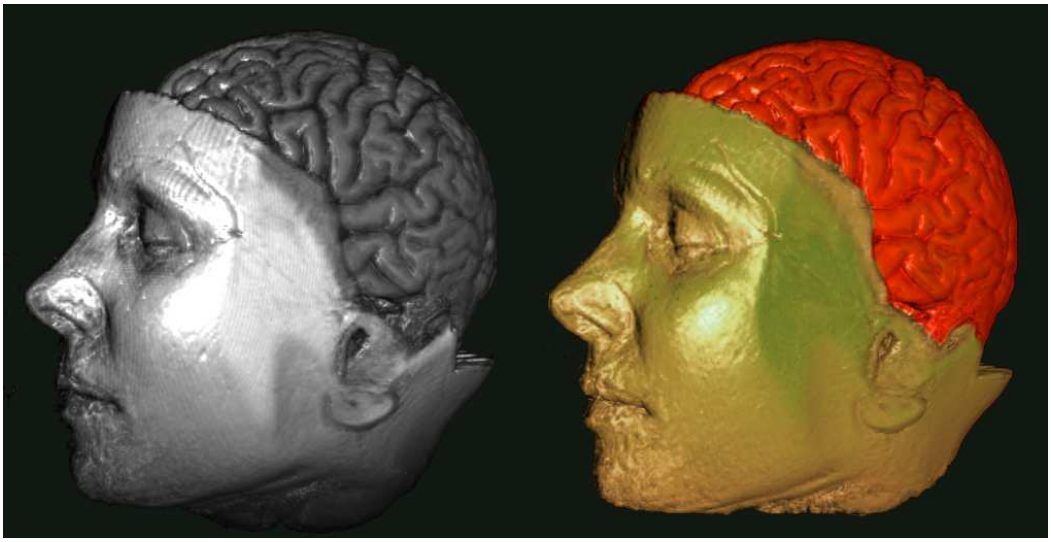


Figure 2.17: The MRI Head dataset [77] rendered with a simple greyscale transfer function (**A**) and with a transfer function that classifies soft tissue and brain tissue (**A**).

visible in 2D slices (A), but can be occluded during volume rendering, depending on camera position and transfer function settings (B). The transfer function can be adjusted to display them, as shown in C, where the skin is hidden and D, where skin is shown as a translucent surface. A reverse of this situation can also occur, when features of interest may be made transparent if the transfer function is not created correctly.

3D visualisation of human CT datasets is seriously affected by occlusion, as most features of interest (for example, suspected tumours, aneurysms, or fractures) lie inside the body. As illustrated in Figure 2.18, internal features usually cannot be displayed without adjusting the transfer function to make most soft tissues transparent or translucent (B), or using clipping planes to cut away parts of the dataset (C). This risks missing important features, and, as stated above, is one of the reasons why 2D slice visualisation is preferred in clinical practice.

2.2.2.4 Illumination

Sampling inside a volumetric dataset and assigning colour and opacity using a transfer function is not enough to generate a realistic rendering of an object. This approach ignores lighting and the role it plays in providing information about the structure and position of objects in 3D computer graphics scenes, as well as in the real world. Therefore, some light emitted by external sources is usually simulated,

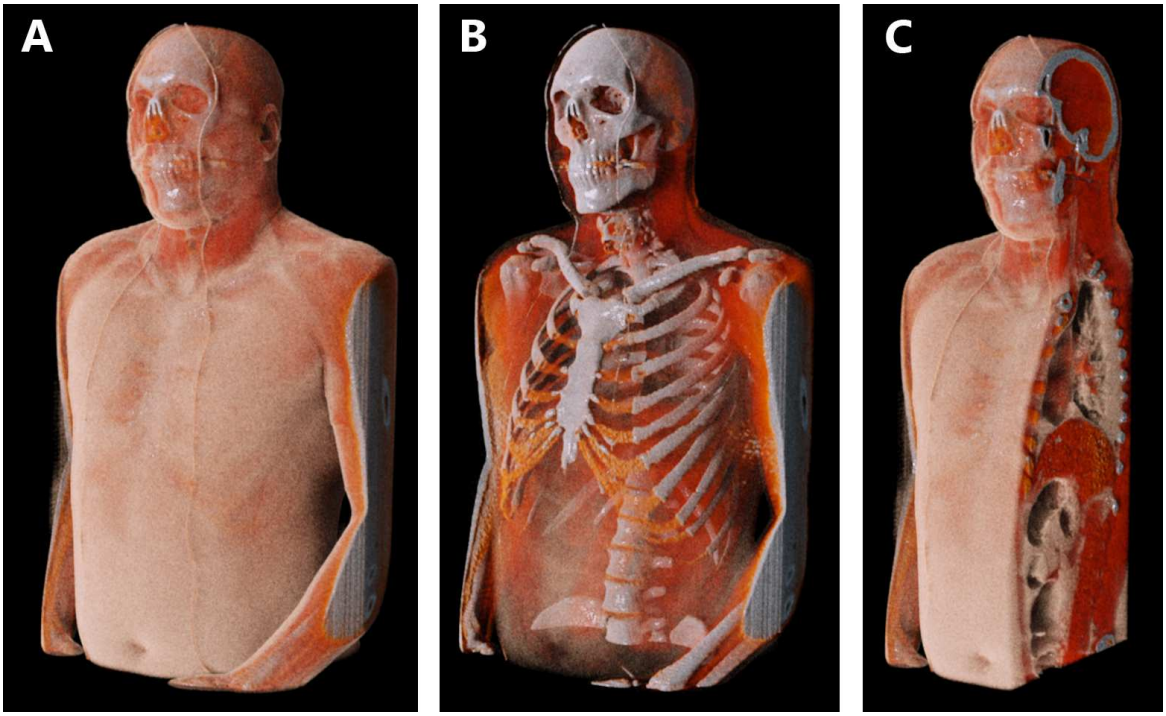


Figure 2.18: **A**: the Visible Human Male dataset (section 9.7) rendered using a transfer function that shows soft tissue as being opaque. **B**: lowering the opacity of soft tissues to show the bones. **C**: using a clipping plane to reveal the detail inside the dataset without lowering the opacity of soft tissues.

and the techniques for doing so are broadly termed "illumination".

Conducting a simulation of light transport through the dataset (complete with scattering and absorption of photons) is the ideal solution, if photorealistic image quality is required. However, for the purposes of visualisation, it is usually infeasible, as the currently-available computing hardware cannot carry out such a simulation at interactive frame rates. Therefore, in practice, an approximate model such as emission-absorption or single scattering is usually used. These techniques are so numerous that they cannot be adequately described in this thesis. Instead, the reader is referred to *Real-Time Volume Graphics* by Hadwiger et al. [60] (chapters 5 and 6) and to a review paper by Jöhnsson et al. [91].

It is enough to mention that, in practice, illumination is usually based on the gradient of the volume at the point of sampling. The gradient is found by using an algorithm such as central differences and treated as the normal vector of an isosurface through that point in 3D space. Once the normal is calculated, it can be used in a simple and efficient shading model such as Blinn-Phong [101]. A high-quality

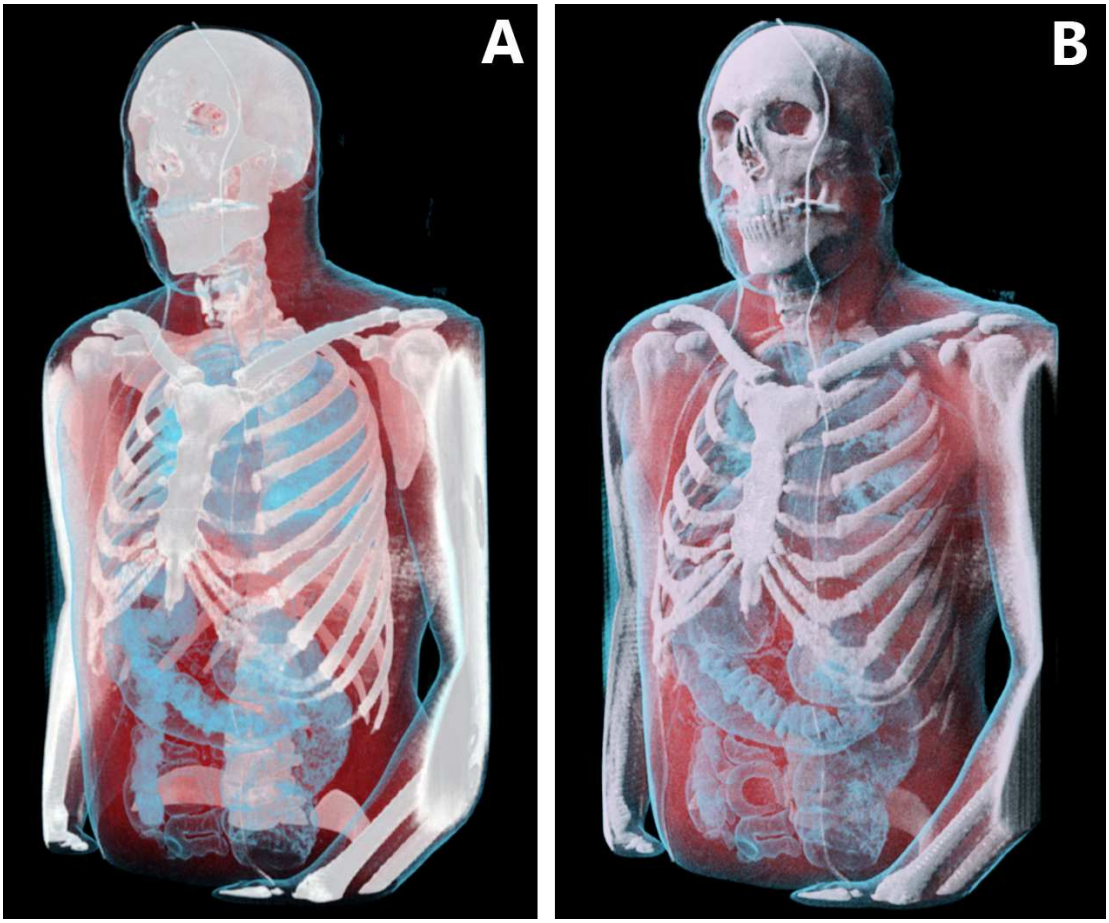


Figure 2.19: The Visible Human Male dataset (section 9.7) rendered without illumination (**A**), and using a physically based illumination model described by Kroes et al. [103] (**B**). The transfer function is the same in both cases.

illumination model can make a substantial difference to the scene by providing depth cues, and can help the observer interpret the information contained in the dataset [102]. Figure 2.19 illustrates the effect of a high-quality physically based illumination model.

2.2.2.5 Compositing and ray termination

The process of combining samples taken by a ray is called compositing. This step determines the contribution of each individual sample to the final colour for a pixel. This contribution is dependent on the maximum number of samples: if the maximum number of samples is high, each sample will contribute less, and vice versa.

A sampling ray terminates if it leaves the bounding box that encloses the volume, accumulates a certain pre-defined opacity (for example, 99% of the maximum possible opacity) or reaches the maximum number of samples, as set by the user.

2.2.2.6 Performance of DVR and acceleration techniques

DVR (or, at least, its naïve implementation) is computationally expensive for two reasons. First, a large portion of the volume must be processed in order for an image to be rendered. A ray marching DVR is essentially a brute-force solution, which samples the dataset millions of times before generating an image. It has no prior knowledge of which regions will be sampled, and which regions will never be sampled by rays. This problem is discussed in more detail in section 5.5.

Second, most DVR algorithms aim for photorealistic image quality and use a high sampling rate and complex illumination techniques to imitate the appearance of the real object as closely as possible. In comparison, 2D slice visualisation usually does not process slice pixels at all, except for applying window and level settings.

Numerous acceleration techniques can be used to increase the performance of a DVR algorithm. These techniques range from reducing the quality settings when the user is moving the camera, to empty-space skipping, which involves classifying regions of a volume as empty or non-empty *prior to* rendering, and skipping empty regions *during* rendering [104]. Another technique is empty space leaping: starting sampling from a certain known point inside the volume, as opposed to starting sampling from a point where the ray intersects the dataset's bounding box. This, in effect, "leaps" across all empty space between the edge of the bounding box and the chosen starting point [105].

These techniques substantially improve the performance of DVR algorithms. For example, Kruger and Westermann [104] report gains of 100-500%, depending on the dataset, while Kainz et al. [106] achieve performance gains of up to 200%.

GPU hardware makes it possible to visualise volumetric datasets at interactive frame rates using DVR. Recent developments in GPU programming (section 2.3) have enabled efficient implementation of volume raycasting DVR algorithms on commodity desktop graphics hardware. GPU-based DVR algorithms are implemented in current state-of-the-art visualisation tools such as Voreen [107,108] and 3D Slicer [68].

2.2.3 Mesh extraction and surface visualisation

Visualisation of surfaces extracted from volumetric datasets is a 3D visualisation technique that relies on an entirely different rendering pipeline to that used in DVR. First, the dataset is pre-processed to generate a mesh based on a certain user-defined threshold: a voxel with a value above the threshold is considered to be solid and a voxel with a value below the threshold is considered to be empty space. The Marching Cubes algorithm [93] is the most common mesh extraction algorithm in use today.

Once a mesh is generated, it can be treated as a standard 3D model. It can be visualised or refined using a 3D modelling or mesh processing tool such as Meshlab [109] or Blender [110], imported into a CAD program to perform measurements and simulations, exported for 3D printing [111], and so on.

Image quality is comparable to that achieved by DVR when the transfer function is set to display a single opaque surface, as shown in Figure 2.20. Two surfaces have been extracted: one showing only the skin of the carp, the other showing only the bones. Note the similarity to images B and C in Figure 2.11.

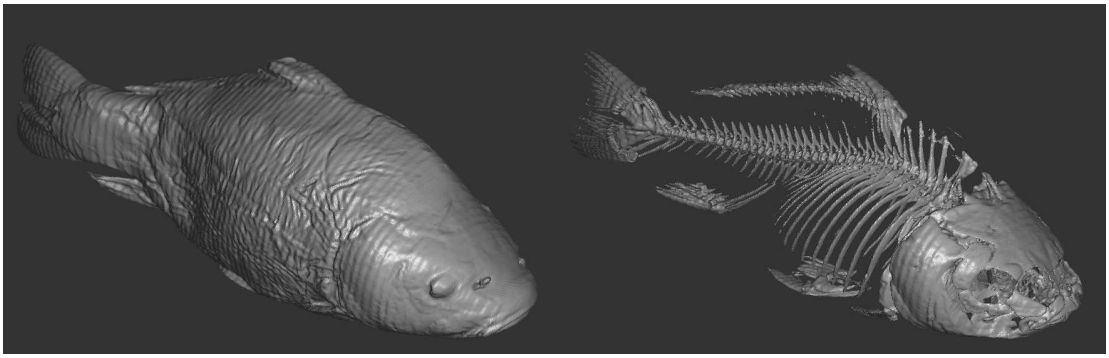


Figure 2.20: Visualisation of the Carp dataset by rendering extracted surface meshes. Left: threshold is set to show skin. Right: threshold is set to show bones.

However, this technique is limited for two reasons:

1. A new mesh must be extracted every time the threshold is updated (although spatial coherence can be used to reduce the extraction time [112]). This is a minor problem, but, depending on the implementation of the mesh extraction algorithm, it may affect interactivity. For example, extracting a mesh from the Carp dataset (shown in Figure 2.20) using a GPU-based

parallel implementation of the Marching Cubes algorithms takes around 26 milliseconds on an NVIDIA Quadro K5200 GPU.

2. Translucent materials, such as liquids and gases, cannot be visualised naturally, by creating the appearance of a material occupying a volume of space. Such materials cannot be rendered properly because mesh extraction only generates a single surface at the boundary between the "empty" voxels and the "solid" voxels. For example, if a body of liquid is visualised, then only its surface will be extracted as a mesh; the inside of the mesh will be hollow.

   Consider the effect in Figure 2.21, which shows soft tissue as a translucent volume of space, while showing bones as an opaque surface. In contrast, extracted surfaces shown in Figure 2.20 cannot be used to achieve the same appearance.

In summary, mesh rendering is a fast technique for visualising a single solid surface. Volume rendering is a better option when a volume of gas or liquid needs to be visualised.

Figure 2.21: Visualisation of the Carp dataset using a transfer function that classifies soft tissue (translucent red) and bones (opaque white). Visualisation of meshes extracted from the same dataset (Figure 2.20) only shows single surfaces.

2.2.4 Multi-variate data visualisation

Multi-variate datasets are datasets that contain measurements of multiple variables. In the case of spectral CT, a single voxel may contain information about several energy and/or material volumes. Therefore, a spectral CT dataset is a special case of a multi-variate dataset, while a single-energy CT dataset is a univariate dataset. Medical imaging technologies such as PET-CT and SPECT-CT also produce multi-variate datasets, as do hyperspectral imaging techniques, which are used for applications such as food quality assessment [113, 114], chemical sample analysis [115], geological exploration [116, 117], and surveillance [118, 119]. This means that we should also examine existing research on multi-variate data visualisation.

The review by Fuchs and Hauser [120] is a good summary of known approaches to multi-variate data visualisation. This review is not directly applicable to spectral CT data visualisation, as the authors do not focus on medical data. Nevertheless, they summarise the existing research and formulate several guidelines that may be applied to the visualisation of any multi-variate data.

- Interacting with the visualisation is essential for exploring complex multi-variate datasets. Actions such as adjusting the colour scheme, viewing the dataset from a different angle, zooming in or out, or selecting a region of interest help the user interpret the data and understand the relationships between the features in the dataset.

- Probing (selecting data points and viewing additional information about them) is highly effective, as it reduces the amount of information presented to the user and allows him or her to study different properties of the same dataset.

- Combining multiple views and different visualisation techniques. For example, a slice of a medical dataset can be displayed using different colour schemes to highlight different features, while additional information about this slice can be shown using graphs, such as scatter plots and histograms [121, 122]. This is done because one visualisation technique may compensate for the deficiencies of another and increase the amount of information presented to the user.

- Colour-coding is a simple yet effective way of visually separating regions of interest and different data types.

- Cutaways or importance-driven rendering [123] in order to focus the user's attention on features of interest. In general, these are referred to "focus+context" techniques because they aim to clearly display a certain region of interest while showing some form of context around it [100]. Similar techniques have also been found to be useful for visualising hyperspectral scientific datasets [124].

- Silhouette and boundary enhancement, which helps the users perceive the location and shape of structures in the dataset [125].

- Automated techniques such as feature classification and segmentation algorithms. Such algorithms may use the information contained in multi-variate datasets (that would normally be too difficult or time-consuming for a human operator to explore manually) and extract features of interest [122, 126].

In general, effective multi-variate data visualisation combines a range of techniques and methods for displaying the data to the user. This is also true of spectral CT data visualisation because energy and material volumes possess different properties, as discussed in section 3.2. Therefore, different techniques may need to be used for displaying and analysing energy and material data. For example, when working with energy volumes, the user typically needs to create a transfer function to visually separate different materials. When working with material volumes, a much simpler transfer function is generally sufficient, as the materials have already been separated. Refer to Chapter 6 for a detailed discussion of transfer function design for spectral CT data visualisation.

2.2.5 Summary

Volumetric datasets can be displayed using a variety of techniques that range from simple slice visualisation to DVR. This section has described three common approaches, which should be viewed as complementary, as each technique possesses certain advantages and disadvantages.

- 2D slice visualisation is fast and simple and requires no special hardware. However, a slice only represents a small portion of the dataset and users must attempt to identify features by navigating through multiple slices. 2D visualisation is commonly used in clinical practice.

- Volume rendering can display entire volumetric datasets, render photorealistic images and use lighting to provide the user with depth cues. However, it requires powerful hardware to achieve interactive performance and usually requires the user to be skilled at designing transfer functions to achieve the desired appearance.

- Mesh extraction and rendering can sometimes be used when a single surface needs to be displayed. It is a very fast technique, but cannot be used to visualise volumes of liquids or gases and translucent materials in general.

2.3 Graphics processing units and GPGPU

This section describes the architecture and capabilities of modern graphics processing units (GPUs) and the design of specialised GPU programming languages. A detailed explanation is necessary because volume rendering algorithms described in this thesis are designed to be executed on GPUs. It is important to emphasise that without a low-cost, easily programmable computing platform such as the GPU, high-quality interactive volume rendering would not be possible.

Traditionally, GPUs have been used almost exclusively to execute standard steps of the graphics pipeline such as vertex transformation and vector and matrix multiplication. However, the GPU is increasingly becoming a standard execution platform for a large variety of algorithms. Algorithms range from matrix multiplication and n-body simulations [127] to fast Fourier transforms (FFTs), volume rendering [74, 106, 128, 129] and video encoding [130]. The expected performance gains, compared to optimised CPU implementations, range from 8x-40x for fast Fourier transforms [131] to 270x for solvers of the sum-product problem [132]. A GPU implementation of an iterative tomographic reconstruction algorithm has yielded gains of 71x for forward projection and 137x for backward projection [133].

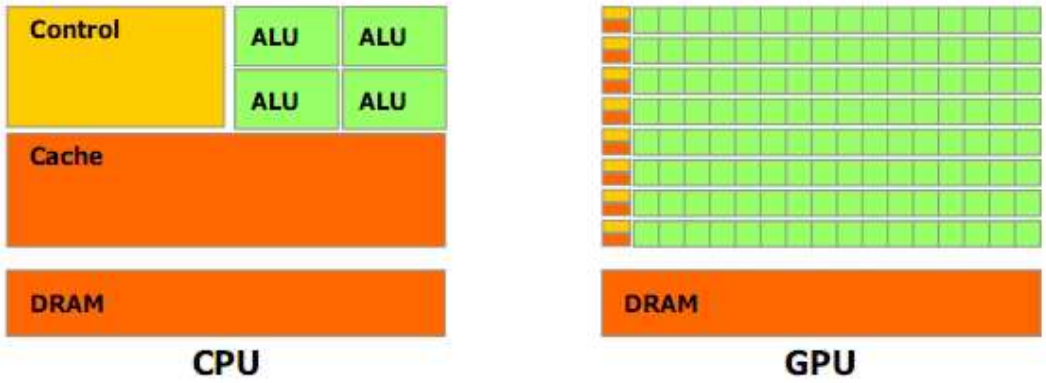


Figure 2.22: Difference between architectures of a typical CPU and GPU. Diagram from the NVIDIA CUDA C Programming Guide Version 4.0 [134].

2.3.1 GPU architecture

The architecture of a typical GPU differs substantially from the architecture of a typical central processing unit (CPU). GPUs are designed to be highly efficient at massively parallel processing of data in the 32-bit single-precision floating-point format. The reason is that most tasks in the standard 3D rendering pipeline, such as per-pixel and per-vertex operations, can be carried out in parallel. These tasks can also be executed on general-purpose processing units such as CPUs, but specialised architectures of GPUs allow for more efficient computation.

On the other hand, GPUs are not suitable for most general-purpose computing tasks that involve a single thread performing a sequence of operations. GPUs have a proportionally higher number of arithmetic logic units (ALUs) and floating-point units (FPUs) compared to CPUs. However, GPU branch prediction hardware is less sophisticated and the amount of cache memory is substantially smaller, as shown in Figure 2.22. Figure 2.23 shows an example of NVIDIA's Fermi, a typical modern GPU architecture.

GPUs are designed for parallel execution of a large number of identical commands on different data elements and exceed CPUs in terms of the maximum theoretical number of operations per second. This is both the key feature, and the main limitation of GPU computing. In theory, GPUs are extremely efficient at parallel computation; in practice, this efficiency is hard to achieve, as supplying a hundreds of execution cores with data is often a limiting factor. Other issues,

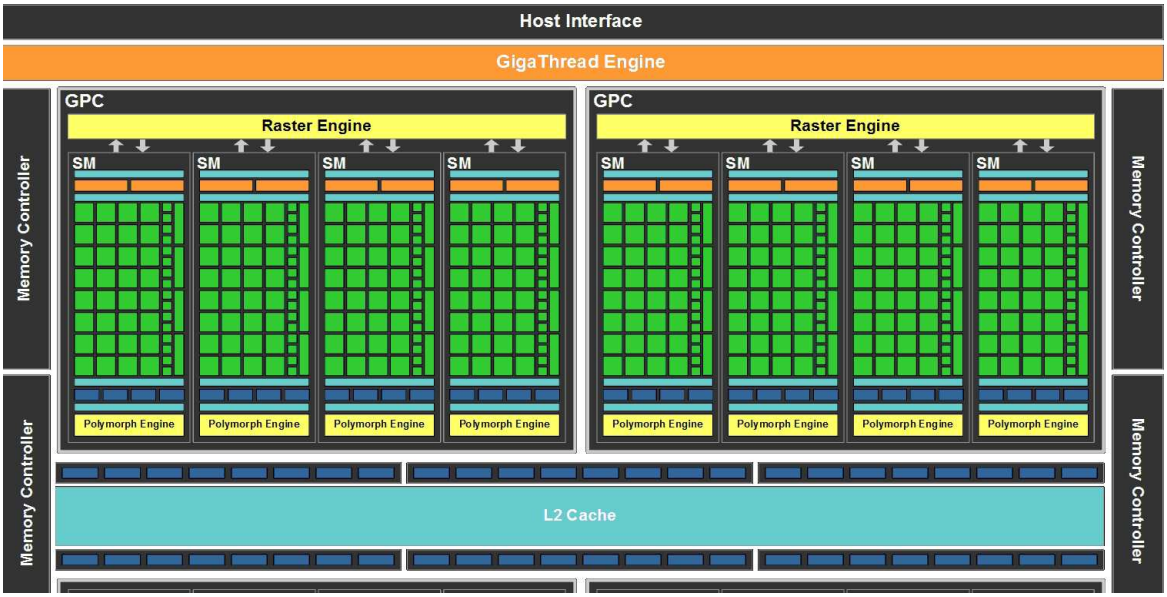


Figure 2.23: Architecture of the Fermi series of graphics cards by NVIDIA [135, 136]. The image shows multiple execution cores grouped into streaming multiprocessors (SMs), along with two levels of caches and specialised auxiliary processing units.

such as branching, also have a proportionally higher impact on GPUs.

2.3.2 Definition of GPGPU

GPGPU stands for General Purpose computing on Graphics Processing Units. It is a technology that enables the programming of GPUs to perform a wide variety of tasks that are not related to graphics processing [137]. This effectively turns an ordinary, low-cost, mass-produced GPU into a powerful co-processor that can execute some tasks faster than the CPU. As a general rule, an algorithm that can be subdivided into a large number of small, independent tasks should benefit from GPGPU acceleration.

Until recently, accessing the computational power of GPUs was difficult: graphics languages OpenGL [138] and DirectX [139] have been used to process and visualise scientific data, but GPU programmability remained limited [104,140,141,142]. These languages are specifically designed to be used for standard 3D graphics rendering tasks, and are not well-suited for scientific computation, which has a different set of requirements.

GPGPU began to gain popularity in 2007, with the release of NVIDIA's CUDA (Compute Unified Device Architecture) [134], followed by the release of

the Khronos Group's OpenCL (Open Compute Language [143]). From that point on, GPU hardware design has been aimed at making the platform more suitable for general purpose computation. Various methods have been used to achieve this, from addition of larger caches and improvements in GPU memory bandwidth.

2.3.3 Volume rendering using GPGPU and CUDA

This section explains how novel features introduced by GPGPU languages such as CUDA have affected the implementation of volume rendering algorithms on GPUs.

GPUs are normally programmed using shading languages such as GLSL [144, 145] or HLSL [146], which are languages specifically designed to access and modify the functionality of a graphics rendering pipeline [147]. Usually, shading languages allow for the implementation of custom vertex transformation, shading and lighting algorithms.

CUDA programming is significantly more flexible than programming with older shading languages largely because any properly declared and allocated array in global GPU memory can be written to. Shading languages are not designed to write to GPU memory, although limited write functionality (for example, writing to frame or vertex buffer objects) is supported. With CUDA, the programmer is given complete control over the allocation and access to GPU memory. Standard features of shading languages, such as textures or hardware-accelerated linear interpolation, are still available.

Version 2.0 of the CUDA SDK (software development kit) [148] included the first official example of a volume raycaster implemented in CUDA. An example is shown in Figure 2.24. Marsalek et al. have improved upon NVIDIA's implementation by optimising threading and memory access patterns, which resulted in a 68-114% performance increase [128].

Kainz et al. [106] have demonstrated some unique advantages of using CUDA for volume rendering by modifying empty-space skipping to make use of CUDA's ability to read from and write to arbitrary buffers in GPU memory. The implementation of empty-space skipping they described updated a GPU-based octree in real-time and achieved gains of around 200% over an unoptimised version. This is an example of how new GPGPU functionality can be used to optimise a volume rendering algorithm. A similar approach is described in this thesis (see section 5.5).

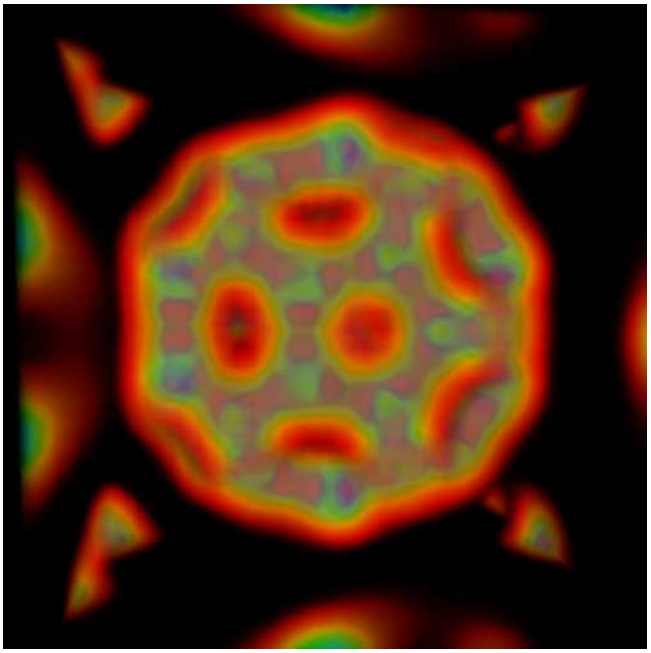


Figure 2.24: The Bucky Ball dataset (32^3 voxels) visualised with a basic volume raycaster from the CUDA 4.0 SDK [149].

Smelyanskiy et al. [150] study the visualisation of medical data using high quality volume rendering algorithms on CPU and GPU architectures and present improved implementations on both platforms. An optimised CUDA implementation was found to be 5-8 times faster than an unoptimised version. It was, however, 12-25% slower than a heavily optimised fragment shader-based volume raycaster that took advantage of hardware rasterisation to perform empty space skipping.

2.3.4 Summary

This section has explained the architecture of modern GPUs and explained the principle of GPGPU. GPGPU is an excellent way to access the resources of commodity graphics hardware. However, implementation must be well-optimised, as unoptimised GPU code is extremely inefficient. Therefore, optimisation must be viewed as an essential part of GPU algorithm design. Despite these limitations, the benefits provided by GPU programming will ensure that it will continue to be used to solve various problems in scientific computing in the foreseeable future.

2.4 Conclusion

This chapter has explored several areas related to the work described in this thesis. In particular, it has focused on the basics of x-ray computed tomography and spectral CT, the techniques for visualising volumetric datasets and the principles of GPGPU. A brief summary of the chapter is given below.

- Spectral CT is a novel medical imaging modality that simultaneously measures the attenuation of photons over several energy ranges of the x-ray spectrum. Several prototype spectral CT systems have been constructed and numerous studies have demonstrated the future clinical applications of spectral CT imaging. The primary advantage of spectral CT is improved material discrimination, which is expected to lead to improved diagnosis of a number of medical conditions including atherosclerosis, cancer, fatty liver disease and osteoarthritis.

- The MARS molecular imaging system is a small animal spectral CT scanner prototype that is based on the Medipix series of photon-counting detectors. The MARS project is working on developing spectral CT technology and investigating the clinical applications of spectral CT imaging.

- There is a large variety of techniques for visualising volumetric datasets generated by medical imaging technologies such as CT. Direct volume rendering is popular because it can generate photorealistic images and can be efficiently implemented on modern GPUs. 2D slice visualisation is a simple technique that needs no special hardware and is commonly used by radiologists and other medical professionals.

- GPUs are designed for massively parallel processing of data and provide an excellent platform for implementing many scientific data processing and visualisation algorithms. Modern GPGPU programming languages such as CUDA greatly simplify GPU programming and allow for more complex algorithms to be developed.

Chapter III

The MARS software toolchain

This chapter explains the structure of the MARS data processing toolchain, the data formats used to store MARS spectral CT datasets and the image quality issues that affect the visualisation of datasets produced by the MARS system.

As described in section 2.1.2, the data acquired by any CT scanner undergoes a large amount of processing before being studied by human users or automatically analysed by software algorithms. In this thesis, the set of applications that performs these tasks is called a data processing toolchain, or simply a toolchain. It usually consists of a set of applications and scripts responsible for tasks such as acquiring projection data during a scan, denoising of projection images, reconstruction of denoised projection images into energy volumes and, finally, extraction of material information from reconstructed energy volumes. Section 3.1 explains the structure and the evolution of the MARS data processing toolchain over the course of this research project.

Section 3.2 describes the two formats currently used to store MARS spectral CT datasets. This section defines these formats, analyses the difference between them, and explains why each format requires a different approach to visualisation.

Finally, the image artefacts present in MARS datasets are described in section 3.3. This section explains how the quality of source data, affects the outcome of 2D and 3D visualisation and why artefacts and noise pose a problem during the visualisation of currently-available MARS datasets.

3.1 Structure of the MARS toolchain

This section discusses the state and evolution of the MARS data processing toolchain between 2012 and 2015. Over these years, the toolchain has been changing gradually, with older scripts and tools constantly being replaced. The trend has been towards greater automation of manual tasks and the conversion of the entire toolchain, from acquisition to visualisation, to conform to the DICOM (Digital

Imaging and Communications in Medicine [65]) standard.

The two versions of the toolchain described in this section have been separated in order to clearly explain the changes in design. The second version should be viewed as a significantly more advanced version of the first, and not as an entirely different set of scripts and applications.

3.1.1 Overview and purpose

The MARS project is developing a complete spectral CT system, which consists of both the MARS scanner hardware and the tools for processing and analysing the data it acquires. Visualisation, the primary topic of this thesis, is only a part of this toolchain. The choice of data processing algorithms, the flow of data through the toolchain and the file formats or network protocols used to exchange data between different programs affect the design of visualisation software. Therefore, at the beginning of my research, my first task was to analyse the structure of the MARS data processing toolchain and suggest improvements.

Figure 3.1 shows the logical structure of the toolchain; that is, the tasks that it performs, as opposed to the specific software that carries out these tasks. First, the MARS scanner acquires a set of projections, which are processed to remove bad pixels (including darkfield masking, statistical tests, thresholds, and user driven masking), converts the data format and corrects for interpixel variation (flatfield normalization), and cleans the data (inpainting, ring filtration, denoising).

Processed projections are then reconstructed using an iterative algebraic reconstruction technique. This results in a single reconstructed CT dataset for each energy bin originally acquired by the MARS scanner. Each dataset of this type is referred to as an "energy volume" (described in detail in section 3.2.1). For more information on this part of the toolchain, refer to the PhD thesis by Niels de Ruiter [59].

After reconstruction, material decomposition (MD) may be performed. This process analyses energy volumes and attempts to classify the materials comprising each voxel. MD produces a number of volumetric datasets, each representing the concentration of a particular material inside the scanned object. Each dataset of this type is referred to as a "material volume" (described in section 3.2.2). This stage of the toolchain is described in the PhD thesis by Christopher Bateman [36].

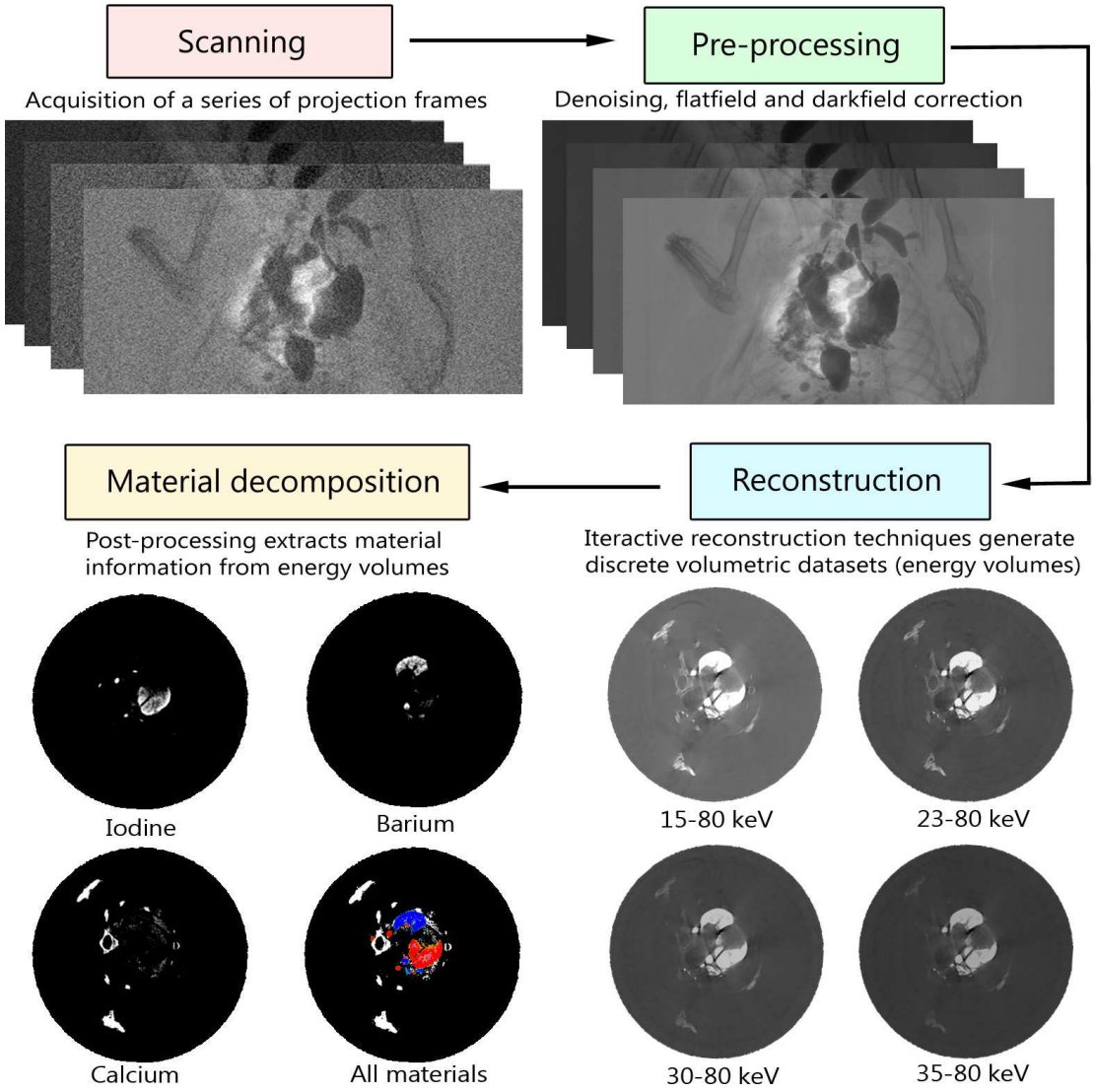


Figure 3.1: The sequence of operations performed by the MARS toolchain. The Mouse12 dataset (section 9.6.1) is used as an example.

3.1.2 Original toolchain design

This is a high-level overview of the toolchain that been used by the MARS project at the beginning of my research. At that stage, the toolchain existed as a collection of applications and scripts with little automation. User input was required for performing most tasks.

Figure 3.2 shows the structure of this version of the MARS toolchain. First, data was acquired by the MARS scanner and automatically uploaded to the PACS (Picture Archiving and Communication System) server. At this stage of develop-

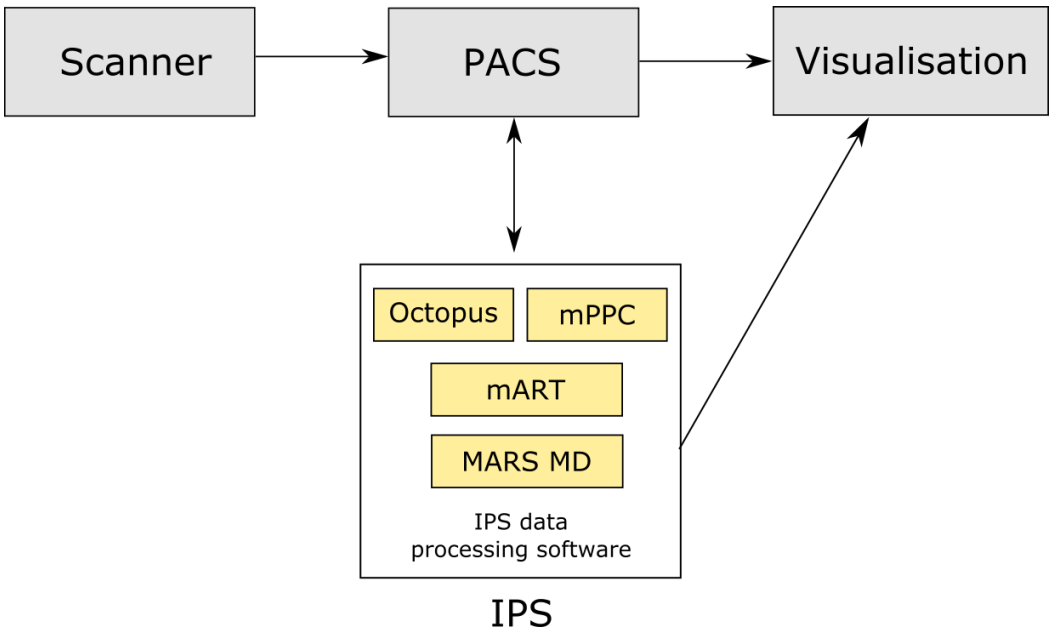


Figure 3.2: Design of MARS toolchain in late 2012/early 2013. The operation of the IPS modules was directly controlled by the user. Data had to be manually transferred between the various components of the toolchain.

ment, the IPS (Image Processing Server) did not exist as a separate application. Instead, it was a name for a collection of image processing and reconstruction applications that were run on a single desktop PC.

There was no underlying framework or protocol that allowed the applications to communicate: the output of one program was saved to disk and used as an input to the next program in the toolchain. The only exception was the communication between the scanner and the PACS server, which used the DICOM networking protocol and initiated the data transfer automatically.

Users carrying out data processing or reconstruction tasks transferred raw projection data from the PACS server to the IPS using a DICOM client such as K-PACS[1] or using portable storage media such as an external hard drive. Preprocessing and reconstruction were performed using the tools created by members of the MARS team, such as mPPC and mART [59], and using commercially available software, such as Octopus CT [152].

In theory, reconstructed datasets could have been manually converted to the DICOM format and uploaded to the MARS PACS server using a DICOM client.

[1] K-PACS, by Merge Software, is no longer available, having been replaced by iQ-VIEW [151]

However, as the responsibility of data management was left to the users, reconstructed energy volumes were often stored on personal devices, such as portable hard drives and flash drives, as opposed to uploading to the PACS server. In addition, material decomposition was performed on a different desktop or laptop PC, and, occasionally, on the Canterbury University BlueFern HPC (high-performance computing) cluster. The lack of a centralised data repository used by all MARS team members often led to difficulties when sharing datasets.

Reconstructed energy volumes and extracted material datasets were stored as stacks of images in the TIFF file format. An experimental mode that output reconstructed datasets in the DICOM format was available, but was not often used. For visualisation, these datasets were manually transferred to special workstation PCs with high-end GPUs. Visualisation and data analysis were performed using custom software such as MARSCTExplorer [129] or general-purpose scientific image processing software such as ImageJ [67] or MATLAB [153].

My contribution to the IPS used in this version of the toolchain includes developing the script to convert DICOM files downloaded from the PACS server to TIFF (Tagged Image File Format [154]) files. This was necessary because, at that time, the scanner suffered from geometric alignment issues, which meant that mART (the custom MARS reconstruction software that supported DICOM files [59]) could not always be used.

Instead, Octopus CT was used to correct the geometric alignment, along with mPPC, which is the pre-processing software for correcting, denoising and stitching projection images. Both applications were only able to load data stored in the TIFF format. Therefore, the raw (unprocessed) projection frames acquired by the scanner had to be converted from DICOM to TIFF according to certain rules.

The script that performed this conversion was called DCM2MPPC and was written in the Python programming language. It formed an essential link between the DICOM data coming from the PACS, and the mPPC and Octopus CT data processing software. However, it also illustrated the need to convert all stages of the toolchain to conform to the DICOM standard.

DCM2MPPC performed the following steps in order to convert projection data into a compatible format:

1. All valid DICOM files in a chosen folder were opened.

2. For each valid DICOM file, the tags, containing information such as the

position of the projection frame, the ranges of all energy bins acquired during the scan, the pixel data format (number of bits per image pixel), and the dimensions of the projection frame, were parsed.

3. The image (which contained the acquired frames for all energy bins, packed into a single pixel array) was partitioned according to the information stored in the tags. For example, if the detector acquired four energy bins at the resolution of 200×200 pixels, then this data would be packed into an array of 200×800 pixels. The offset for the data from the first energy bin would be (0,0), the offset for the second bin would be (0,200), and so on.

4. The frames for all energy bins were saved as separate TIFF images in appropriate folders.

The projection images in the TIFF format were then pre-processed with mPPC, before being loaded into Octopus CT, which allowed the user to manually find the correct geometric alignment parameters and reconstruct the dataset.

In conclusion, the first version of the MARS toolchain was functional, but suffered from serious usability problems. In particular, data transfer between machines had to be initiated manually and the sharing of data between team members was inconvenient. Some software applications that formed the IPS had to be linked together by scripts for converting data. Material decomposition was experimental and was not routinely performed. Nevertheless, the MARS team has used this toolchain for numerous pre-clinical studies into the applications of spectral CT [4, 9, 11, 13, 155].

3.1.3 New toolchain design

At the end of 2012, I and several members of the MARS development team have identified the disjoint nature of the first version of the toolchain as a major problem and an obstacle for further development. In order to be effective for scientific and clinical use, the components of the MARS toolchain had to be integrated and automated as much as possible.

The new design described in this section corrects most of the problems identified in section 3.1.2 and automates most image processing tasks. This design, shown in Figure 3.3, emphasises the importance of connecting all components by using

the DICOM standard. This includes using the DICOM file format to store all raw, processed and reconstructed images and using the DICOM networking protocol to communicate between different machines. The implementation of this design is not yet complete. This is reflected in the diagram in Figure 3.3, which makes a distinction between the existing and the planned toolchain modules.

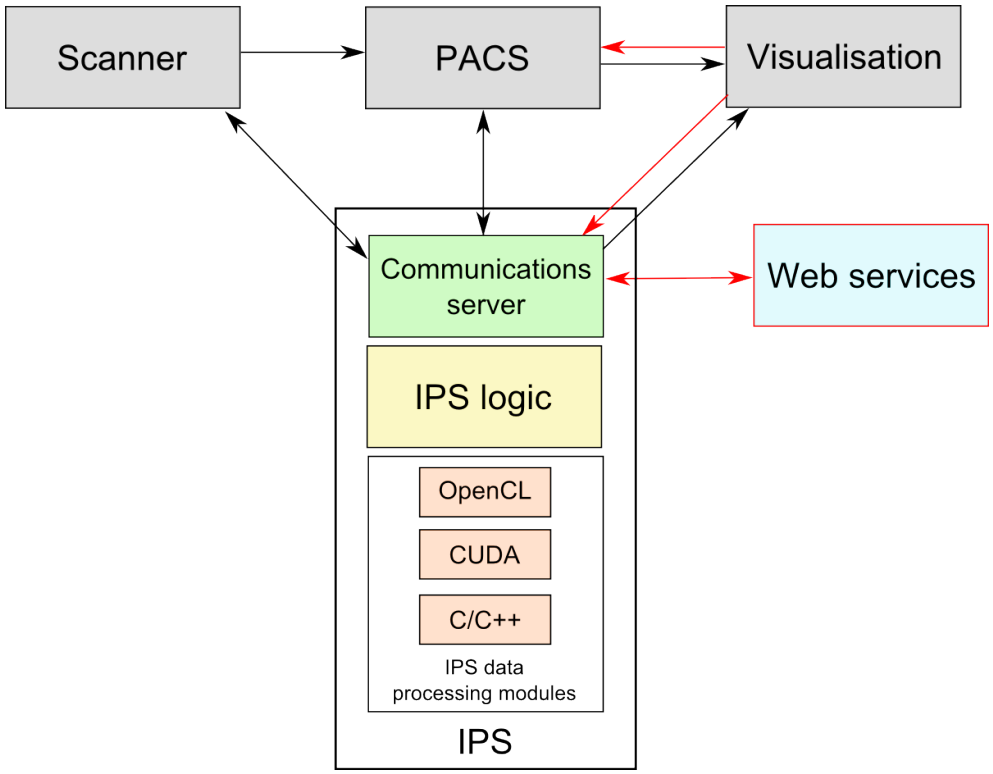


Figure 3.3: New design of the MARS software toolchain. Red arrows indicate the communication between the modules that has not yet been implemented. Red outlines indicate the components that have not yet been implemented.

Aside from helping to design the architecture of this toolchain, my contribution consisted of developing most of the visualisation algorithms and some data processing tools. These are described in Chapters 5, 6, 7 and 8.

This section describes several major changes introduced in the new MARS toolchain that have the greatest impact on the user's workflow. The first change is the transition to the DICOM standard. DICOM is a comprehensive standard that is widely used in clinical settings. It includes the specifications for file storage and transfer, as well as the Unified Worklist and Procedure Step (UPS) [156], which can be used for controlling DICOM-compatible software remotely. For example,

the UPS allows one application to initiate a sequence of data processing tasks by submitting a list of steps that need to be performed. Other UPS-compatible clients can receive this worklist and carry out the steps it describes.

The second major difference is the structure of the IPS. The IPS is now partitioned into several components: the communications layer that uses the DICOM networking standard to accept connections and transfer data, the logic layer that distributes the work across the available hardware, and the data processing modules that are called by the logic layer.

This design is modular and decouples the transfer of data between the IPS and the other workstations from the low-level image processing tasks. The users access the functionality provided by the IPS logic layer indirectly, by querying the communications layer.

The communications layer is responsible for handling incoming and outgoing network connections. Different modules can be added to it as the number of nodes (such as scanners, PACS servers, web browsers or visualisation clients) that can communicate with the IPS increases. The exact architecture of this layer has not been finalised.

The IPS logic layer is designed to balance the load on the available hardware by assigning different tasks to the most suitable hardware platforms. For instance, easily parallelisable tasks, such as denoising algorithms applied to individual slices of a reconstructed energy volume can be carried out by GPU algorithms. Complex, sequential tasks such as iterative algebraic reconstruction, can be assigned to a CPU. This is a departure from the previous toolchain architecture where the choice of processing software was left to the user. The new approach only allows the user to choose the *tasks* to be performed; the IPS logic layer decides how to carry them out.

This loose coupling of IPS modules allows them to operate independently and leaves open the possibility of running some algorithms on HPC hardware (in addition to the standard CPU and GPU platforms). HPC data processing algorithms can be implemented in a separate IPS data processing module, which would be an optional component that does not require any changes to the overall toolchain architecture.

The third major difference is the integration of visualisation with the rest of the toolchain. Users are able to use the visualisation client to query the PACS server and access the list of available datasets. This eliminates the manual transfer of

data to the visualisation workstation that was necessary when using the previous toolchain.

Currently, the MARS toolchain only supports the transfer of data from the PACS server to the visualisation client. As of August 2015, the transfer of data in the other direction has not been implemented. However, in the future, the users will be able to modify datasets (for instance, by annotation or segmentation) and upload the changes back to the PACS server using the visualisation client.

Finally, a web browser-based interface for accessing the functionality of the IPS (referred to as "web services" in Figure 3.3) is planned for this version of the toolchain. This module has not yet been implemented. The web interface is intended to allow the users to issue data processing commands and view raw, processed or reconstructed data from any compatible browser. In the future, this interface may replace the custom applications that are currently required to issue data processing commands to the IPS.

3.1.3.1 Performance of the IPS software

The IPS machine currently used by the MARS project has dual Intel Xeon CPUs (48 cores in total) and 64GB of DDR3 RAM. Usually, 5 energy volumes are reconstructed per scan, and 6 material channels are identified. For a large dataset such as a mouse (approximately $670 \times 670 \times 1000$ voxels), processing takes around 20-30 minutes. A smaller dataset such as a calibration phantom (approximately 400x400x300 voxels) will take around 5 minutes to process.

3.1.3.2 Workflow examples

As stated above, the implementation of the new MARS software toolchain is not yet complete. However, in the future, this toolchain design will support use cases such as the ones described below.

- The communication between the scanner and the IPS can be used to generate a live preview of the scanned object. This may be a small set of low-quality slices that is quickly generated and displayed to the operator in order to help adjust the scan parameters.

- The communication between the visualisation client, the PACS server, and the IPS, can be used to perform data processing initiated by the visualisation

client. For example, if a dataset contains artefacts that have gone undetected during pre-processing and reconstruction, the user is able to send a command to the IPS to apply denoising or perform artefact removal. The IPS will choose the appropriate algorithms and hardware platforms to perform these tasks and upload the modified dataset to the PACS server. At the same time, it will inform the visualisation client that the modified dataset is available, which will allow the client to automatically download and display it.

This task would be much more time-consuming if the first version of the MARS toolchain was used. The dataset would need to be transferred manually to the IPS, denoising would need to be performed manually, and the modified dataset would need to be transferred back to the visualisation workstation.

3.1.4 Summary

The MARS software toolchain is a set of applications used for acquiring, processing and visualising MARS spectral CT datasets. The toolchain has undergone significant changes over the past three years. The early version consisted of a collection of applications that required manual configuration and continuous user input. I have significantly improved this toolchain by analysing its structure, suggesting enhancements, and creating an essential link between two of its stages. This work is still ongoing, with the toolchain currently being redesigned to increase automation, reduce the user's workload, and use the DICOM standard for network communication and file storage.

3.2 MARS spectral CT data types

This section discusses the current state of MARS spectral CT data processing and describes the two different formats used to store MARS datasets. Originally (before the development of MARS MD algorithms [36]), MARS spectral CT datasets consisted of a set of energy volumes. Recent advances in material decomposition have led to the creation of a second data type. This section describes both types, as detailed knowledge about the nature of the reconstructed data produced by the MARS toolchain is the first requirement for developing the algorithms and software to visualise it.

3.2.1 Energy volumes

The energy volume is a standard data type in all CT imaging. It is a volumetric dataset where voxels are positioned on a regular grid in 3D space. Voxel values represent the linear attenuation coefficients of x-rays over a certain energy range. Therefore, a conventional single-energy CT dataset is an energy volume [157]. However, it must be noted that in clinical practice linear attenuation coefficients are nearly always converted into Hounsfield Units (section 3.2.1.1).

The MARS system can measure up to 8 different energy ranges, and, consequently, can process and reconstruct up to 8 energy volumes from a single scan. When visualising an energy volume, users attempt to identify the attenuation ranges that correspond to materials or organs of interest. This is usually done by using a transfer function (described in section 2.2.2.1) to classify attenuation ranges and attempt to visually separate various structures or organs. All energy volumes contain all materials and organs of interest; however, some have better contrast than others.

Therefore, visualising a single energy volume is the same as visualising a conventional CT dataset. Doing so does not take full advantage of the properties of spectral CT. In particular, spectral CT improves material discrimination, but material information cannot be easily accessed and understood by users during the visualisation of a single energy volume.

As an example, consider Figure 3.4, which shows a slice from two energy volumes of a spectral CT dataset of a mouse (Mouse12, described in detail in section 9.6.1). This dataset contains an iodine- and a barium-based contrast agent, along with other materials, such as water in the soft tissues and calcium in bones [6]. However, when visualising energy volumes separately, it is not immediately clear how these materials may be extracted or quantified.

There are slight differences in attenuation coefficients between the volumes, which is a result of measuring different energy ranges (Figure 3.4C). However, the differences are small, and the contrast of the difference image needs to be enhanced in post-processing to clearly visualise them (Figure 3.4D).

In some cases, materials can be identified by intermixing the data from several energy volumes during interactive visualisation. However, it is a task that requires a high degree of proficiency in transfer function design and the use of specialised tools [129].

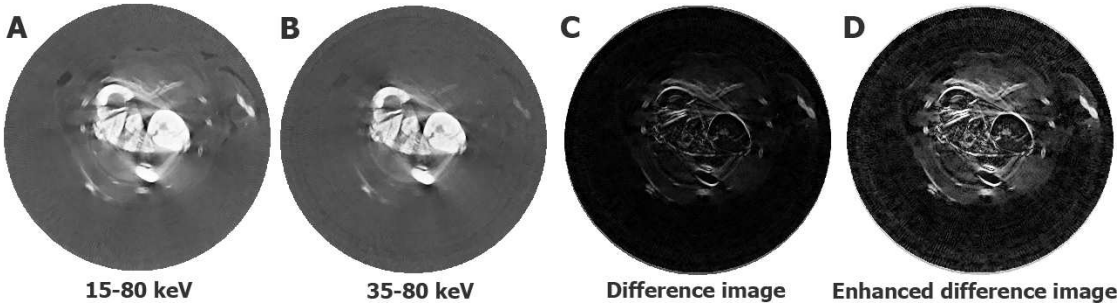


Figure 3.4: **A**, **B**: slice from two energy volumes of the Mouse12 dataset (section 9.6.1). **C**: a subtracted image. **D**: a subtracted image processed to clearly show the differences.

In practice, such intermixing is not commonly performed by members of the MARS project due to the extreme difficulty of the process. Currently, this approach is considered to be outdated and unnecessary, as better solutions, such as post-reconstruction material decomposition (section 3.2.2) have been found.

However, in limited circumstances, the research conducted using the MARS system requires energy volumes to be studied without further post-processing such as MD. An example is shown in Figure 3.5. This dataset is a medical hip joint implant made of a cobalt and chromium alloy, used for testing beam hardening artefact reduction (further discussed in section 3.3) using the MARS system.

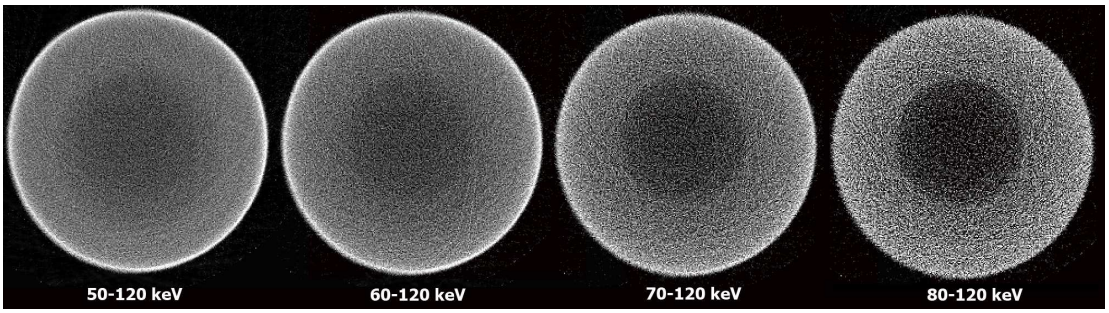


Figure 3.5: A slice of the CoCr Ball dataset (section 9.5.2). The differences between the energy volumes are much clearer than in Figure 3.4. The difference depends on the composition of the object and on the chosen energy bin ranges.

Displaying slices of energy volumes separately demonstrates that beam hardening affects broad energy bins (for example, the 50-120 keV bin) more than narrow energy bins (80-120 keV). For example, in the 50-120 keV bin, the hollow space inside the implant is indistinct; however, it is clearly visible in narrower, higher-energy bins.

In conclusion, energy volumes may be visualised separately, or combined during visualisation. Currently, every study using the MARS system produces energy volumes, while MD is only performed in some cases, due to the difficulty of calibrating the algorithms, as described in section 3.2.2.

3.2.1.1 Hounsfield Units

Hounsfield Units (HU) relate the linear attenuation coefficient of a substance to the linear attenuation coefficient of distilled water at standard pressure and temperature. Each voxel of the dataset is transformed according to the following formula:

$$\mathrm{HU} = 1000 \times \frac{\mu - \mu_{\mathrm{water}}}{\mu_{\mathrm{water}} - \mu_{\mathrm{air}}} \tag{3.1}$$

where μ is the linear attenuation coefficient of the voxel and μ_{air} and μ_{water} are the measured linear attenuation coefficients of air and water.

The Hounsfield scale has no upper limit, but in clinical practice materials range from -1000 HU (the attenuation of air) to around 3000 HU (the attenuation of dense bone), although very dense materials, such as surgical implants made of steel, can have higher HU values.

Hounsfield Units are the most common data format in clinical practice for two reasons. First, water and air are used as reference materials, which creates a very convenient scale for human imaging, as the majority of soft tissues are mostly composed of water. This scale is suitable for medical professionals who may not be familiar with the physics of x-ray imaging. Second, CT datasets acquired using different scanner hardware can be standardised and scaled to the same range.

The MARS project does not normally use convert linear attenuation coefficients to Hounsfield Units. However, this conversion can be performed, if necessary. The reason why this conversion is not routinely performed is that Hounsfield Units are energy-dependent [158]. If two different x-ray spectra were used to measure the attenuation of the same substance, and the attenuation coefficients were converted to Hounsfield scale, then the HU values would be different. This means that the linear attenuation coefficients of water and air must be experimentally obtained for each energy, which requires additional work, as the energy bin ranges vary depending on the scan protocol. MARS MD uses linear attenuation as the input, so HU conversion is not required.

However, for the purposes of visualisation, the units used to store voxel values in energy volumes are irrelevant, as the transformation to Hounsfield Units does not correct the biggest problem associated with energy volume visualisation. Each energy volume still contains all materials or tissues, which must be manually identified and separated by the user.

3.2.2 Material volumes

A set of energy volumes is one possible way of representing an object scanned by the MARS system. The second option is representing an object as a set of materials. Each material is stored in a separate volumetric dataset, where each voxel stores a certain concentration value. This dataset is called a material volume.

Examples of materials currently identified by the MARS MD algorithms include water-like and fat-like materials [4], iodine- and barium-like materials [6], and iron- and gold-like materials [55]. This naming convention is used because the MD process is based on comparing signals to an expected pattern. For example, any voxel that appears to match the water pattern better than the other patterns will be classified as water, regardless of its actual content.

For brevity, this thesis will not use this convention, instead referring to identified materials simply as *water*, *calcium*, *barium*, and so on. However, the *-like* postfix is always implied when referring to MARS material volumes. For example, the proper name for a volume of iodine identified by MARS MD algorithms is an *iodine-like* volume.

In the context of spectral CT MD, the term *material* refers to any substance with a distinct mass attenuation profile, usually a chemical element or a molecule. This is the reason why the MARS system can be used for *molecular imaging*, which refers to the targeting of specific molecules. For convenience, material volumes are also sometimes referred to as *channels*: for example, the *calcium channel* or the *barium channel* shown in Figure 3.6.

As mentioned in section 2.1.2, materials can be separated using multivariate analysis techniques, such as principal component analysis. The current MARS MD algorithm is using a modified basis material decomposition technique [36]. This algorithm extracts material information from reconstructed energy volumes by comparing the measured linear attenuation coefficient for each voxel to the known attenuation profiles of a number of materials.

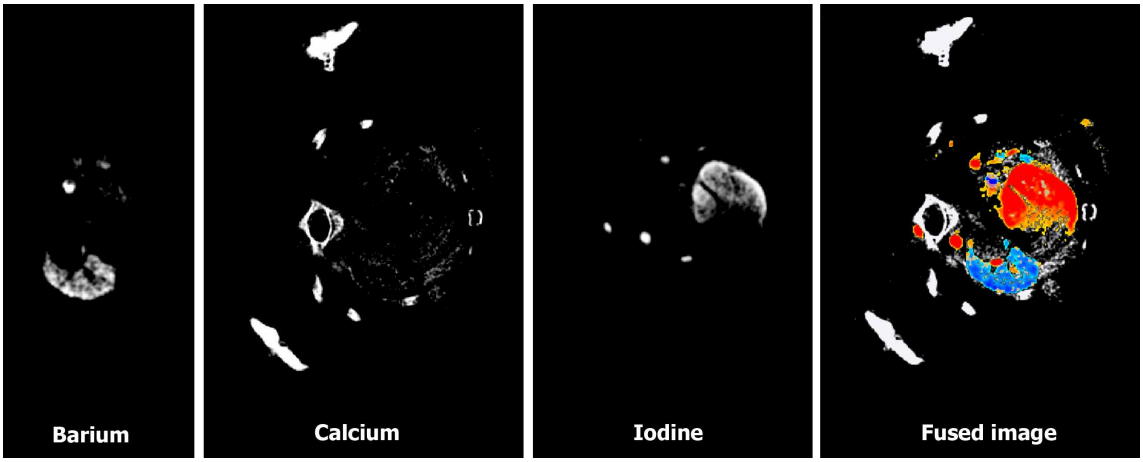


Figure 3.6: Material decomposition of the Mouse12 dataset (section 9.6.1). Three materials are first shown separately, then in a single fused image to show their relative positions inside the body of a mouse. Iodine: yellow (lower concentrations) to red (higher concentrations). Calcium: white. Barium: cyan (lower) to blue (higher).

Direct reconstruction of material volumes from projection images (*simultaneous material reconstruction*) is also possible [33], and is currently being investigated by the MARS team. Therefore, it must be emphasised that material volumes are not merely an extension of the energy volume data type. Material volumes can be reconstructed independently, and, in fact, an inverse process can be carried out to transform material information into energy information (that is, a set of energy volumes can be created from material volumes).

However, the exact methods of producing material volumes are not important for the purposes of this research. This section focuses on the data format itself, and on the different approaches to visualisation that are required when working with material volumes.

One example of material data produced by the MARS system is provided in Figure 3.6, which shows the same slice as Figure 3.4. However, it is immediately clear which materials are present in this slice. Another example is shown in Figure 3.7. The iodine and calcium present in a scan of an excised section of a human tibia have been separated and quantified.

Visualising and analysing material volumes is easier for three reasons:

- In an energy volume, a single voxel may contain multiple materials. However, a known linear attenuation coefficient or HU value is not necessarily sufficient to separate these materials and to determine their concentration.

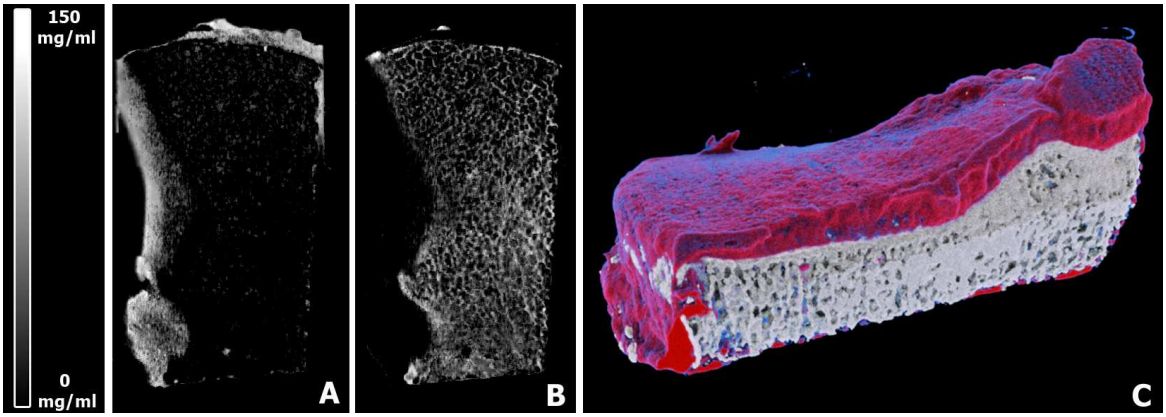


Figure 3.7: Two material volumes of the Knee Cartilage dataset (section 9.4). **A**: a slice of the iodine volume. **B**: a slice of the calcium volume. **C**: volume rendering of both volumes. Calcium: grey (lower concentrations) to white (higher concentrations). Iodine: blue (lower concentrations) to pink (higher concentrations).

Due to noise from various sources, materials with similar energy responses cannot always be distinguished. Consequently, the user cannot always create a transfer function that separates different materials. This will be explained in greater detail in Chapter 6.

In contrast, MD automatically classifies every voxel in the dataset. This means that (assuming a perfect MD algorithm) overlapping materials in a single voxel will be assigned to separate volumes. During visualisation or analysis, the user can choose to only work with certain materials of interest and ignore all other materials.

- Voxel values are stored in commonly-used units (mass concentration, most commonly g/ml), as opposed to specialised units (linear attenuation coefficients or Hounsfield Units). This presents data in a format easily understandable by users who may not be familiar with x-ray physics or medical imaging.

- Measurements are simpler to perform. For example, a certain amount of a contrast agent can be injected, and the amount that has reached the regions of interest (ROIs) can be measured directly.

The most serious disadvantage associated with material datasets is that they do not contain the full anatomy, or structure of the scanned object. For example,

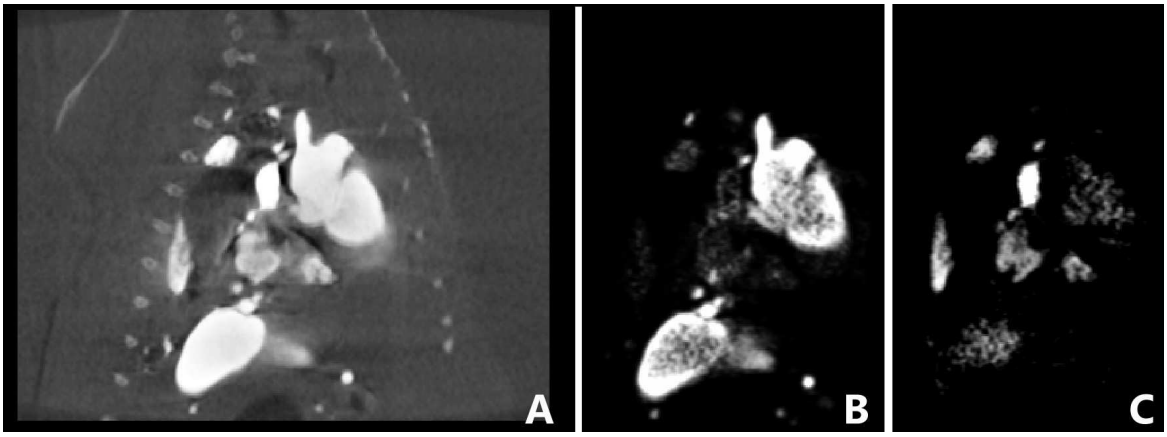


Figure 3.8: **A**: slice of an energy volume of the Mouse12 dataset (section 9.6.1). **B**: corresponding slice of the iodine volume. **C**: corresponding slice of the barium volume.

compare the information contained in a slice of an energy volume of the Mouse12 dataset in Figure 3.8A to the material information shown in Figure 3.8B and C. The iodine and barium channels only contain these materials, which, in theory, makes interaction significantly easier, as the users do not need to separate them manually. However, these channels also provide no context necessary to visualise their location in relation to the other anatomical features. In contrast, the energy volume shows the skeleton of the mouse and the soft tissues, as well as the barium in the lungs, and the iodine in the heart, although the task of separating them is left to the user.

Continued development of MARS MD algorithms [36] suggests that material data will become standard in the future, and that every MARS scan will contain several materials in addition to multiple energy volumes. As a consequence, this thesis focuses on a number of tools that are intended to be used primarily for working with material volumes, and on the fusion of material and energy information.

3.2.3 Summary

This section has explained the difference between the two data formats currently used to store MARS spectral CT datasets. There are two reasons for emphasising the differences between these data formats:

1. Energy volumes represent the attenuation of x-rays inside the object. Material volumes represent the concentration of materials inside it. These are

two fundamentally different ways of representing the same physical object.

2. The data type determines which visualisation and interaction techniques are suitable for working with it. This distinction is most obvious when designing transfer functions, as discussed in Chapter 6.

It is important to emphasise that each energy or material volume can be considered a standalone volumetric dataset, and can be visualised and analysed separately. However, doing so may not take full advantage of the additional information provided by spectral CT.

All information required to separate materials is present in energy volumes, as well as in the original projection images. The difficulty lies in extracting this information and presenting it to users. A high degree of technical competence and familiarity with computer graphics algorithms is required in order to extract material information from energy volumes during visualisation. In contrast, material decomposition creates volumetric datasets that are much simpler for the users to work with.

In conclusion, MARS spectral CT data exists in two different formats. Both formats are volumetric datasets, and can be visualised using the same basic 2D and 3D techniques. However, custom tools are usually required.

This thesis focuses on material volume visualisation and the fusion of energy and material data. Chapter 6 shows that custom GUIs can facilitate transfer function design for material datasets and Chapter 7 explains the design of custom tools for simplifying interaction with multi-volume datasets. Finally, Chapter 9 describes the use of a combination of these tools for visualising spectral CT datasets obtained during pre-clinical research with the MARS system.

3.3 Image quality and image artefacts

This section demonstrates the effects of image artefacts on the visualisation of MARS spectral CT datasets. Understanding the impact of artefacts and noise is necessary because poor image quality has an adverse effect on the outcome of visualisation. In other words, these defects make it harder for users to study the features and composition of MARS datasets.

Visualisation is normally the last step in any medical or scientific data processing toolchain: at this stage, datasets are ready to be analysed by users. Therefore,

the outcome of visualisation depends not only on the algorithms and tools used, but also on all the data processing carried out prior to it.

Detailed investigation of the nature and exact causes of various CT image artefacts is not within the scope of this thesis. Therefore, a brief summary is provided below, while interested readers are referred to an in-depth review by Boas and Fleischmann [159].

Image artefacts are a persistent problem in all variants of x-ray CT. Most artefact types are common to single-energy, dual-energy and spectral CT, as they are caused by the same sources, such as:

- Malfunctioning detector elements. Such elements can produce abnormally bright or dark pixel in the projection images [23]. Reconstruction algorithms combine these pixels and produce bright or dark "rings", as discussed later in this section.

- Mechanical defects such as the misalignment of parts inside the scanner, which can produce a range of artefacts, depending on which part is misaligned.

- Motion artefacts, which typically result in a blurred image. This is a problem in clinical imaging, where patients can inadvertently move during a scan. The MARS team does not often scan sedated live animals, so motion artefacts in MARS datasets are usually due to the sample changing shape during a very long scan (several hours).

- Noise due to an insufficient number of photons reaching the detector. The signal produced by the detector always contains some amount of Poisson noise, and its effects are more pronounced at lower photon counts [159]. Increasing the exposure time will improve the signal-to-noise ratio, but will also increase the dose to the patient. Clearly, this is undesirable, so clinicians try to adhere to the ALARA (as low as reasonably achievable) principle [160], which aims to minimise the radiation dose while maintaining reasonable image quality. Therefore, some degree noise is acceptable in clinical CT imaging, as it is important to use the shortest exposure that still allows for accurate diagnosis.

- The physical limitations of x-ray imaging. A good example is *beam hardening*, which produces dark streaks near highly-attenuating objects (for exam-

ple, metal implants or bones). It is caused by the uneven absorption of x-rays of different energies by human body tissues. However, not all reconstruction algorithms take this property of x-ray imaging into account; instead, the x-ray beam is assumed to be monochromatic (composed of photons of the same energy). This artefact type is further described in section 9.5, which discusses the use of the MARS system for metal implant imaging.

- Scatter. X-ray photons are scattered by the tissues they travel through, with highly-attenuating tissues or objects causing more scatter. This can change the direction of a photon (it no longer travels in a straight line from the source to the detector), reduce its energy, or change both of these properties at once [161].

Some defects may also be introduced by image processing algorithms. Figure 3.9 shows the artefacts present in the Visible Human Male dataset, acquired by a commercial single-energy CT system (General Electric High Speed Advantage [162]).

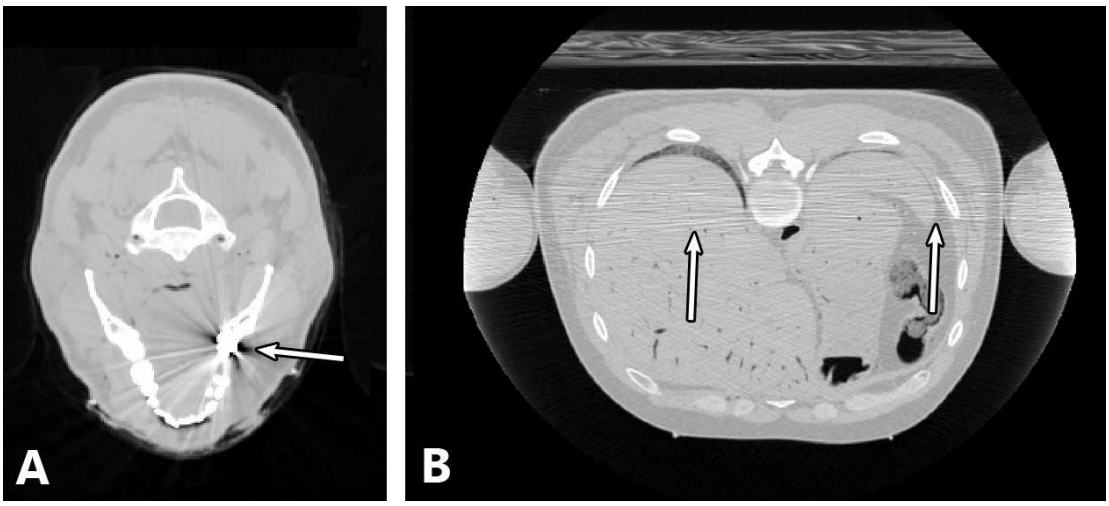


Figure 3.9: Beam hardening artefacts (**A**) and streak artefacts (**B**) in two slices of the Visible Human Male dataset (section 9.7.1). The artefacts are marked by arrows.

Finally, some artefacts are specific to the photon counting detectors used in many prototype spectral CT systems such as MARS. For example, charge sharing [163] results in speckle noise and misclassification of voxels by MD algorithms.

It must be noted that image artefacts are a problem that, in the ideal case, should be corrected *prior to* visualisation. However, this section shows that, at

the current stage of development, the MARS data processing toolchain does not remove all artefacts from energy and material volumes. Therefore, special tools may sometimes be required to mitigate the effects of noise and image artefacts *during* visualisation.

Image artefacts are present at all stages of the MARS toolchain. Various defects contained in projection frames are translated into energy volumes by reconstruction algorithms. Next, some artefacts may be transferred into material volumes by the MD algorithms.

The MARS team has conducted some work on denoising and filtering of spectral CT datasets. De Ruiter [59] has worked on denoising of MARS datasets and Rajendran et al. [9] have investigated the reduction of beam hardening artefacts using the MARS system. However, most available MARS datasets are still affected to some degree.

This thesis explores both 2D and 3D visualisation techniques. Therefore, it is important to describe the appearance of artefacts in individual slices of a volume and during 3D rendering of the entire volume.

Figure 3.10 shows some typical CT artefacts as they appear during 2D slice visualisation of MARS datasets:

(A) Beam hardening in scan of a titanium screw inserted into a piece of bone (TiScrew dataset, section 9.5). The part of the screw inside the bone appears to have lower attenuation than the part protruding from the bone. This is a result of beam hardening and not a real physical property of the screw.

(B) Ring and streak artefacts in an energy volume of a mouse dataset (section 9.6).

(C) Ring artefacts in the water channel of the Meat1127 dataset (section 9.3). This image shows the translation of artefacts present in energy volumes into material volumes.

Next, Figure 3.11 shows that the artefacts present in slices of a volumetric dataset directly translate into 3D space during volume visualisation. The same three datasets are shown:

(A) Beam hardening visualised through the use of colour. Bone is shown in grey. The attenuation range corresponding to metal is assigned a gradient from

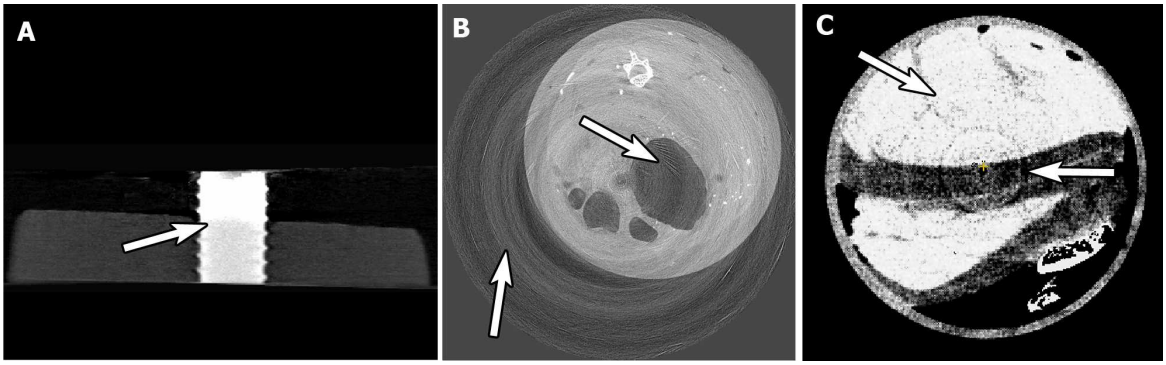


Figure 3.10: Artefacts in slices of spectral CT datasets acquired by the MARS system. Artefacts are marked by arrows.

red (lower attenuation) to blue (higher attenuation). A close-up of the screw is shown in an inset below the image, and a clipping plane is used to show detail inside the screw. The visible colour gradient reflects the difference in the measured attenuation, which is a result of beam hardening.

(B) 2D streak artefacts visible in Figure 3.10B are also clearly visible during 3D visualisation. These artefacts can be seen on the left and right of the mouse's spine.

(C) Ring artefacts in Figure 3.10C are also visible as rings during DVR of the water channel of the Meat1127 dataset.

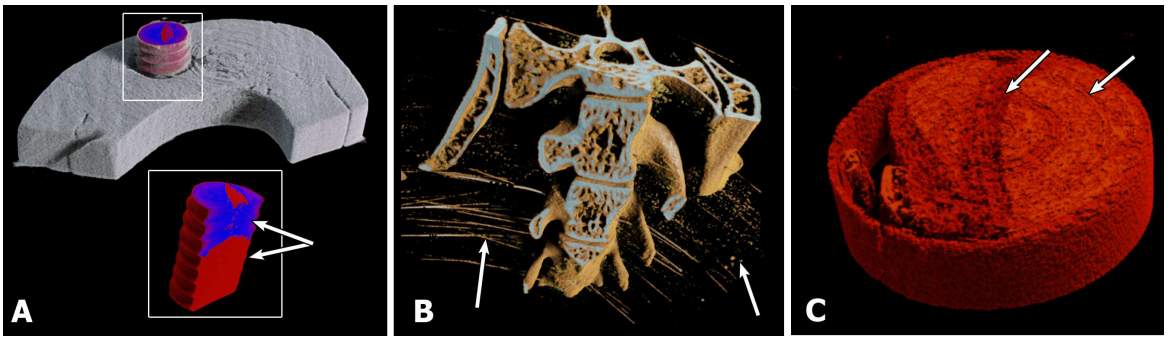


Figure 3.11: Artefacts shown in Figure 3.10 as they appear during 3D visualisation. Visible artefacts are marked by arrows.

Figure 3.11 shows that, during 3D visualisation, artefacts (streaks and rings in

particular) may occlude the regions of interest. Currently, this is one of the major problems facing users attempting to visualise MARS datasets.

The artefacts shown in in Figures 3.10 and 3.11 clearly degrade the quality of images presented to the user. However, some datasets are affected so severely that entire regions are rendered unusable and ROIs are occluded. One such case is shown in Figure 3.12.

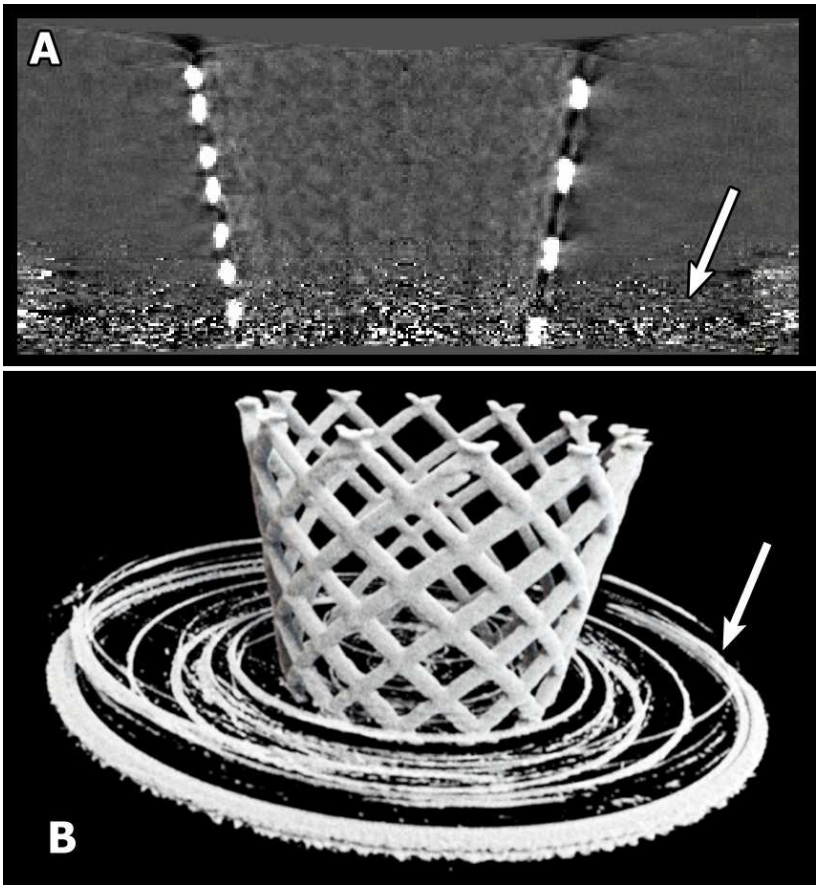


Figure 3.12: Ring artefacts visible during 2D and 3D visualisation of the TiMesh dataset (section 9.5.1). The speckles of noise visible in the lower part of the 2D slice appear as large, prominent rings during DVR. The ring-like appearance is due to the noise following a certain pattern, which is caused by faulty detector elements that report incorrect photon counts.

The artefacts shown in Figures 3.10, 3.11 and 3.12 are clearly visible and some can be removed or suppressed at the visualisation stage. However, the effects of some artefacts are subtle and may not be immediately obvious to the user.

For example, Figures 3.10B and 3.11B show beam hardening as it appears

during the visualisation of energy volumes. However, beam hardening can also affect MD because it distorts the measured attenuation of voxels. This can lead to incorrect classification or quantification of materials by MD algorithms. Therefore, the users should be aware of this possibility, even though it is not a problem that can be solved at the visualisation stage.

3.3.1 Summary

This section has described the appearance of various image artefacts during 2D and 3D visualisation. Currently, most MARS spectral CT datasets contain numerous streaks and rings, as well as beam hardening artefacts that affect both visualisation, and analysis.

Therefore, the visualisation algorithms and tools designed for working with MARS datasets must take these known image quality problems into account. This thesis proposes the use of advanced data interpolation algorithms (section 5.3.3.1) and interactive volume data editing and removal tools (section 8.5) to reduce the effects of noise and image artefacts.

3.4 Summary

This chapter has explained the contribution of this research to the design of the MARS software toolchain, described the formats currently used to store MARS spectral CT datasets, and explained the appearance of the artefacts that these datasets often contain.

- My early work has contributed to the design of the MARS software toolchain, and, in particular, to the concept of an image processing server composed of multiple independent components. The visualisation software and the tools developed over the course of this research (Chapters 5-8) are integrated into this toolchain.

- MARS spectral CT datasets are stored in two different forms: energy volumes and material volumes. The former stores the attenuation of x-rays of different energies inside the object, while the latter represents the concentrations of materials inside the object. Current MARS spectral CT datasets can consist of a combination of these data types.

- Currently, both energy, and material volumes produced by the MARS software toolchain contain image artefacts and noise that are not fully removed during pre-processing. These artefacts distort the appearance of the dataset during visualisation, occlude regions of interest, and affect the accuracy of material decomposition.

Chapter IV

Requirements for MARS spectral CT data visualisation

This chapter reviews the current state of medical data visualisation, examining several modalities, such as dual-energy CT, PET-CT, MRI, and spectral CT. It examines all aspects of visualisation, from 3D rendering algorithms to graphical user interface (GUI) design. This information, along with the knowledge gathered from the review of the current state of the MARS toolchain, data formats and image artefacts (conducted in Chapter 3), is used to summarise the requirements for effective visualisation of datasets acquired during pre-clinical research with the MARS molecular imaging system.

Spectral CT is currently undergoing a transition from being a purely experimental technology, to a technology that is close to being introduced into clinical practice [3]. Scientific researchers constitute the majority of current users of spectral CT. However, medical professionals are expected to comprise a significant portion of the user base in the future. Therefore, this chapter evaluates the needs of both scientists, and clinicians, but mostly focuses on the requirements of the researchers currently working with the MARS system.

This chapter is structured as follows:

- Section 4.1 conducts a brief review of the techniques currently used to visualise the data produced by functional imaging modalities, such as positron emission tomography (PET) and functional MRI (fMRI). These modalities have been chosen because the functional information is similar to the material information produced by spectral CT, and the techniques used for visualising this information are closely related to those used for spectral CT data visualisation.

- Section 4.2 examines the state of dual-energy CT data visualisation. Dual-energy CT is the closest imaging modality to spectral CT in terms of its capabilities, the acquisition hardware and the data processing algorithms.

Therefore, it is important to review the techniques used for visualising the data generated by this modality.

- Section 4.3 describes the methods used for visualising spectral CT data by the MARS team and the other spectral CT research teams. It also outlines the deficiencies of the tools available to the MARS team before the beginning of my PhD project and explains why further research into the tools for visualising and analysing MARS datasets was required.

- Finally, section 4.4 uses all information gathered by the reviews in Chapter 3 and sections 4.1-4.3 to draw conclusions about the requirements for effective visualisation of MARS spectral CT datasets.

4.1 Visualisation of PET-CT and MRI/fMRI datasets

This section briefly reviews the methods used for visualising data produced by PET-CT, which is an imaging modality that shares many similarities with spectral CT, and functional MRI (fMRI), which is a well-established medical imaging technology for measuring the changes in neural activity over time. PET-CT and fMRI were chosen because they are good examples of commonly-used functional imaging modalities.

As the name implies, functional imaging provides the information about the *function* of certain organs or regions of the body. Usually, functional imaging is concerned with measuring the metabolic activity inside the body. In contrast, single-energy CT only captures the anatomy, which is the *structure* of the body. Dual-energy and spectral CT also capture the anatomy, but also allow for much more precise material discrimination (that is, the assessment of the molecular composition of tissues, which is also called molecular imaging [164]).

4.1.1 PET-CT data visualisation

PET-CT is a technology that employs a combination of a PET (positron emission tomography) scanner and a CT scanner. PET enables functional imaging by providing information about the metabolic activity in the body [165]; however, it does not capture the patient's anatomy as well as CT, and has lower spatial resolution. Therefore, a combination of both modalities is beneficial.

To measure metabolic activity, PET employs radiopharmaceuticals: compounds that are injected into the patient and undergo radioactive decay. The most commonly-used radiopharmaceutical is [^{18}F]fluorodeoxyglucose [166] (FDG). As it decays, a chain of nuclear reactions produces photon pairs, which are detected by the PET scanner. Since FDG is a glucose analogue, more of it is used by the cells with a high metabolic rate than by the cells with a lower metabolic rate. Therefore, greater rates of positron emission will be observed from regions with high metabolic activity. An example is shown in Figure 4.1B.

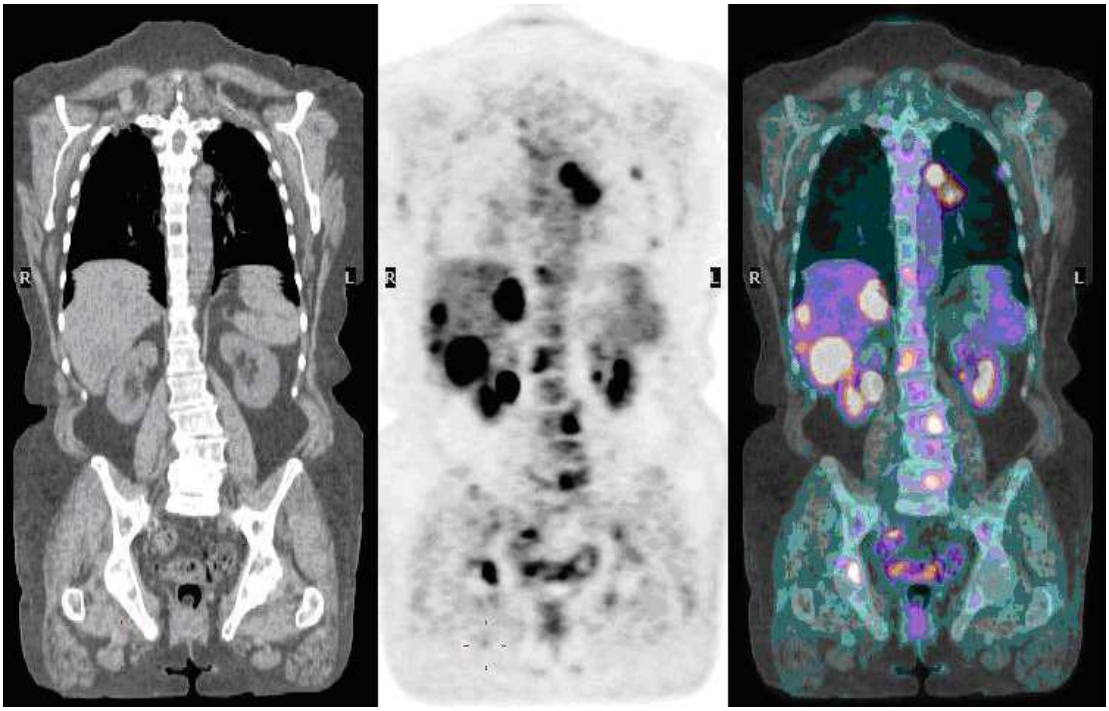


Figure 4.1: Visualisation of a slice of a whole body PET-CT scan. Left: CT. Middle: PET, with regions of high metabolic activity shown in black. Right: fused PET-CT image, with areas of high metabolic activity shown in white.

In clinical imaging, highly active regions correspond to organs such as the brain or the heart, as well as to malignant tumours, where the cells are dividing rapidly. PET clearly shows these regions, but does not always provide sufficient anatomical information, which may be needed to associate them with a particular organ.

For example, consider the CT and PET images in Figure 4.1A and B, respectively. The CT image contains the complete anatomy of the patient, showing multiple organs and tissue types. In contrast, the PET image only shows the

regions where positron emission was observed, which may not necessarily correspond to particular organs. Fusing CT and PET images (Figure 4.1C) allows the observer to find the regions of high metabolic activity, *and* visualise them within the anatomical context.

This is the reason why PET-CT has been demonstrated to improve diagnostic effectiveness in comparison to PET and CT being used separately [167]. In particular, it improves detection and staging of various type of cancer: for example, prostate cancer [168], hepatocellular carcinoma [169], sarcoma [170], and lymphoma [171].

The standard approach to PET-CT data fusion involves displaying corresponding slices of PET and CT datasets side-by-side (Figure 4.1A-B and Figure 4.2A), or fusing them together (Figure 4.1C and Figure 4.2B). The CT image, having higher spatial resolution, is used as a background, while PET data is overlaid in colour. This is the most common clinical visualisation technique, used by Buck et al. [167], Pfannenberg et al [172], Mortensen et al. [173], Koolen et al. [174], Adams et al. [171], and many others.

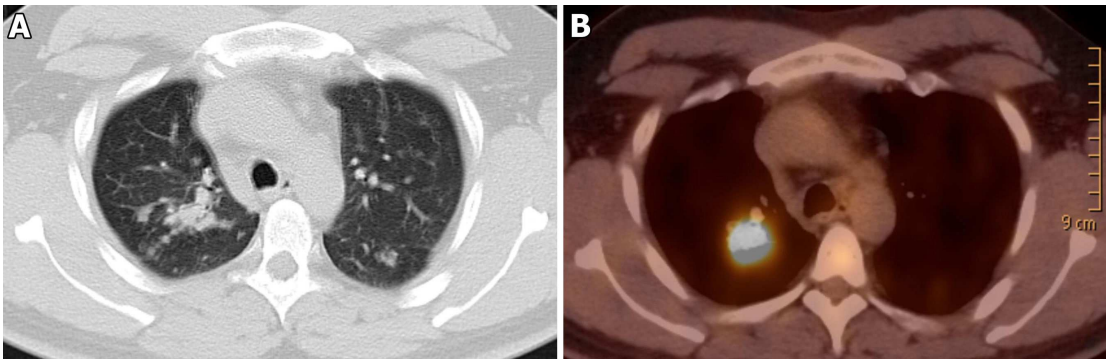


Figure 4.2: **A**: CT image of the lungs affected by sarcoidosis [175]. **B**: fused image, with PET data being overlaid onto the CT image. The area of high metabolic activity is clearly visible.

3D visualisation of PET-CT data has also been studied before. For examples Kim et al. [176] and Yung et al. [177], use DVR to render both PET, and CT data concurrently. In both studies, the same concept was used, with CT data being shown in greyscale, and PET data being assigned a bright colour scheme. An example is shown in Figure 4.3. Finally, the software provided by commercial vendors such as Siemens is also capable of 3D rendering of PET-CT data [178].

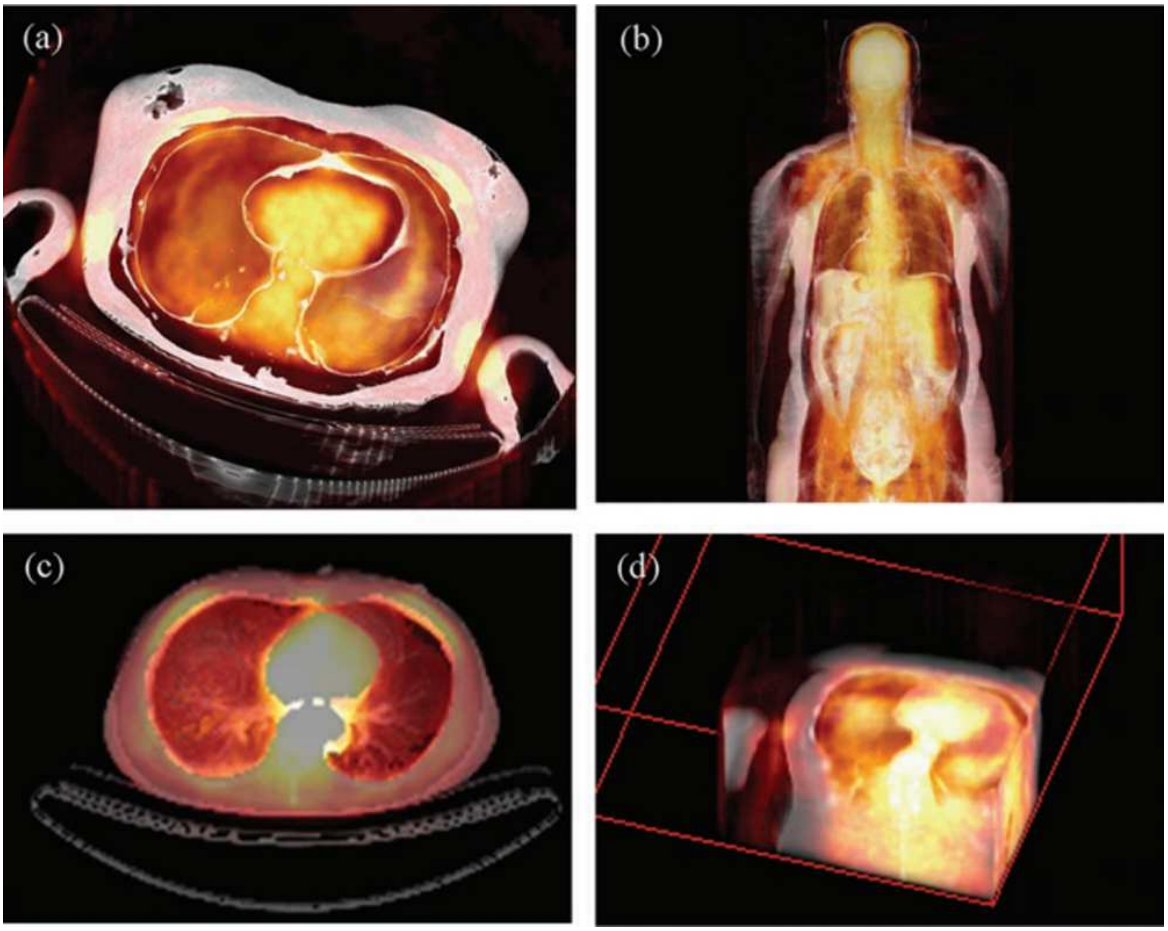


Figure 4.3: Direct volume rendering of PET-CT data, with segmentation being used to enhance the lung boundaries in **A** and **C**. Figure from: J. Kim et al. Real-Time Volume Rendering Visualization of Dual-Modality PET/CT Images With Interactive Fuzzy Thresholding Segmentation. IEEE Transactions on Information Technology in Biomedicine, 2007, 11, 161-169.

In summary, knowledge of the techniques used for PET-CT data visualisation is valuable, as it shows how the metabolic information may be placed within the anatomical context. The same may also need to be done during spectral CT data visualisation (with the metabolic information being replaced by the material information): as discussed in section 3.2.2, the material information extracted from energy volumes is not always useful by itself; some form of context may need to be shown alongside it.

However, the standard visualisation technique for PET-CT data cannot be directly applied to spectral CT data. The reason is that PET-CT generates two

datasets (one conventional CT, and one PET dataset) that must be fused. In contrast, the material detection capabilities of spectral CT are not limited to two materials: for example, the current version of the MARS scanner is able to identify and quantify up to seven [36]. This number is likely to increase as the detector capabilities and data processing algorithms improve in the future. Therefore, the concept of energy and functional data fusion is directly applicable to spectral CT, but the exact visualisation methods used in PET-CT imaging are not sufficient.

4.1.2 MRI and fMRI data visualisation

Magnetic resonance imaging, or MRI, uses a strong magnetic field to align the magnetic moments of protons. Pulses of radio waves are then used to excite the protons, which, in turn, emit radio frequency photons as they return to their standard states. The frequency required for excitation depends on the strength of the magnetic field and on the atom being imaged. For example, in a magnetic field with a strength 1 Tesla (T), pulses of 42.58 MHz are required to induce resonance in hydrogen atoms (hydrogen, due to its abundance in the human body, is the most commonly-imaged atom). For a detailed description of MRI technology, refer to Chapter 15 of *The Essential Physics for Medical Imaging* by Bushberg et al. [20].

Two types of signals, known as T1 and T2 relaxation times, are typically measured [179]. The relaxation time varies depending on the properties of the tissue in which the hydrogen atoms are found, the strength of the magnetic field, and the temperature of the object. For example, at 37° C, and 1.5 T, the T2 relaxation time for muscle is around 35 ms, while for cartilage it is 42 ms [180]. Therefore, tissues can be discriminated based on their T1 and/or T2 signals.

This sensitivity makes MRI excellent for detecting small variations in the composition of soft tissues[1]. This modality is used for a number of diverse clinical applications, such as:

- Atherosclerotic plaque imaging [181]
- Stroke and oedema detection [182]
- Study of brain morphology in patients with conditions such as schizophrenia [183] and autism [184]

[1] This is beyond the capabilities of single-energy CT, and is one of the primary reasons for the development of spectral CT.

- Cancer imaging [185, 186]
- Determining the effects of ageing on brain structure [187]
- Ischemic heart disease imaging [188]

In clinical settings, MRI data visualisation is largely restricted to the same techniques as single-energy CT. Slices are usually visualised using a greyscale colour scheme [183, 187, 189, 190, 191], as shown in Figure 4.4. The colour, or shade of grey, represents a particular tissue type, which varies based on the scan parameters. Unlike CT, the brightness of the colour does not necessarily correspond to an increase in the density of tissue.

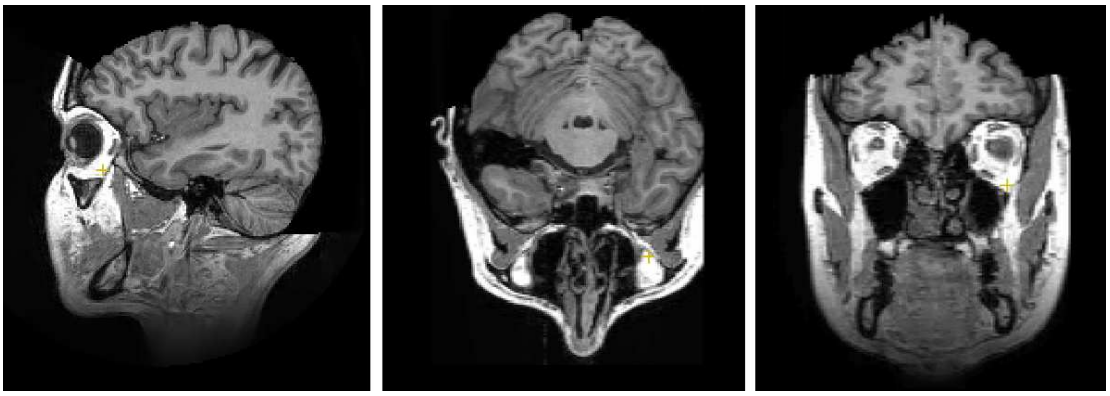


Figure 4.4: Visualisation of slices of a standard MRI dataset (MRI Head [77]). Soft tissues are clearly displayed, while bone is invisible. Further, different types of soft tissue are discriminated: for example, brain tissue appears different from the soft tissues around the face.

For the purposes of this review, the most interesting and relevant type of MRI imaging is functional MRI (fMRI). It tracks the changes in neural activity over time by measuring the differences in the oxygenation of blood (called a blood-oxygen-level-dependent, or BOLD, signal [192]): active areas consume more oxygen and produce a different signal. Rapidly acquiring a series of images allows for these changes to be tracked and subsequently visualised. Depending on the system, the temporal resolution ranges from hundreds of milliseconds [193] to seconds [194].

fMRI data is usually been displayed by overlaying an activity map on top of a standard high-resolution MRI image [20]. In clinical practice, this overlay is nearly always restricted to 2D slice visualisation, as shown in Figures 4.5 and 4.6. For

further examples, refer to the studies by Knutson et al. [195], Winterer et al. [196], Rueckert et al. [197] and Ishai et al. [198].

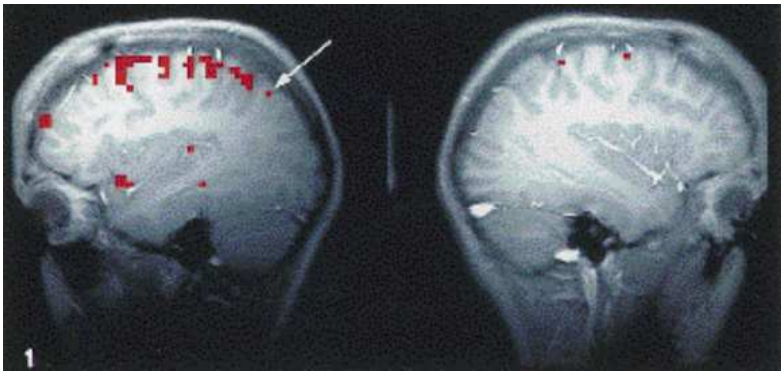


Figure 4.5: Visualisation of the neural activity (shown in red) detected by fMRI. A slice of a standard high-resolution MRI dataset is used as a context. Figure from: L. Rueckert et al. Visualizing cortical activation during mental calculation with functional MRI. NeuroImage, Volume 3, Issue 2, April 1996, Pages 97-103. Used with permission from Elsevier B.V.

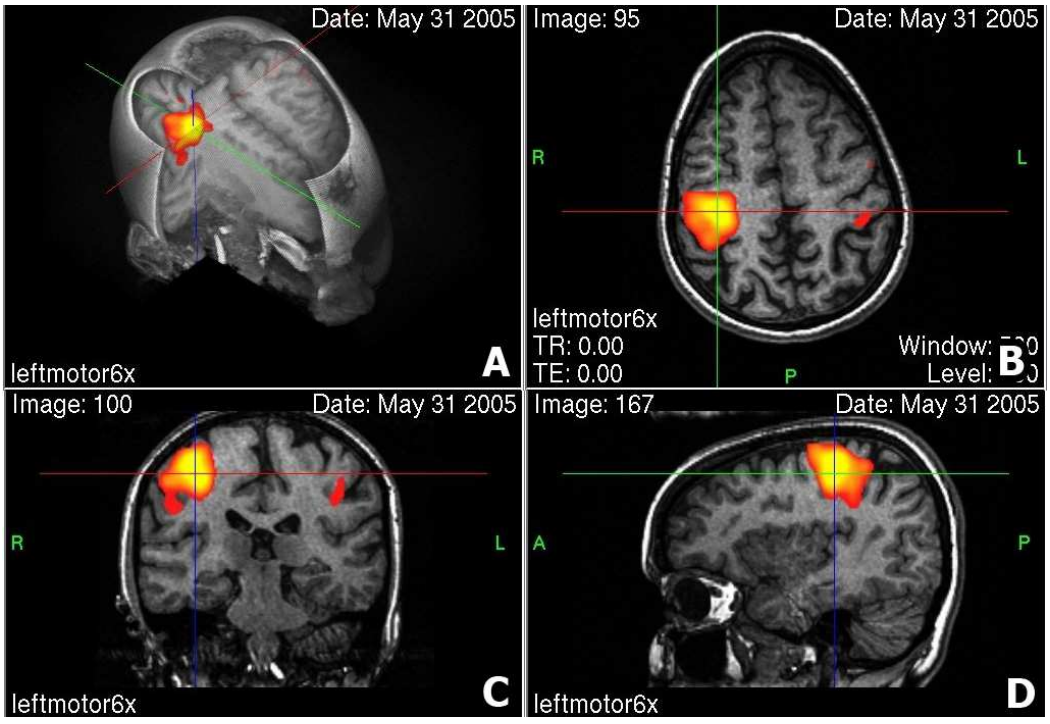


Figure 4.6: Visualisation of the neural activity detected by fMRI. The activity map generated by fMRI is shown in colour, while the anatomical context (provided by a conventional MRI image) is shown in greyscale. Both 3D (**A**), and 2D (**B**, **C**, **D**) techniques are used. Image released into the public domain by Martin White.

3D visualisation of fMRI data is similar to 3D visualisation of PET-CT data, as both must provide the context for the functional information they present. Roessler et al. [199] developed a DVR framework for visualising fMRI data, using a template model of a human brain (not acquired during the same scan) for context. Archip et al. [200], investigated 3D visualisation and segmentation of MRI and fMRI data for neurosurgery, and Eklund et al. created an interface for analysing and visualising connectivity maps (maps of correlation between the activity in different regions of the brain) [201]. In all cases, functional information (fMRI data and/or connectivity maps) were displayed along standard MRI datasets, which functioned as anatomical context. An example is shown in Figure 4.7.

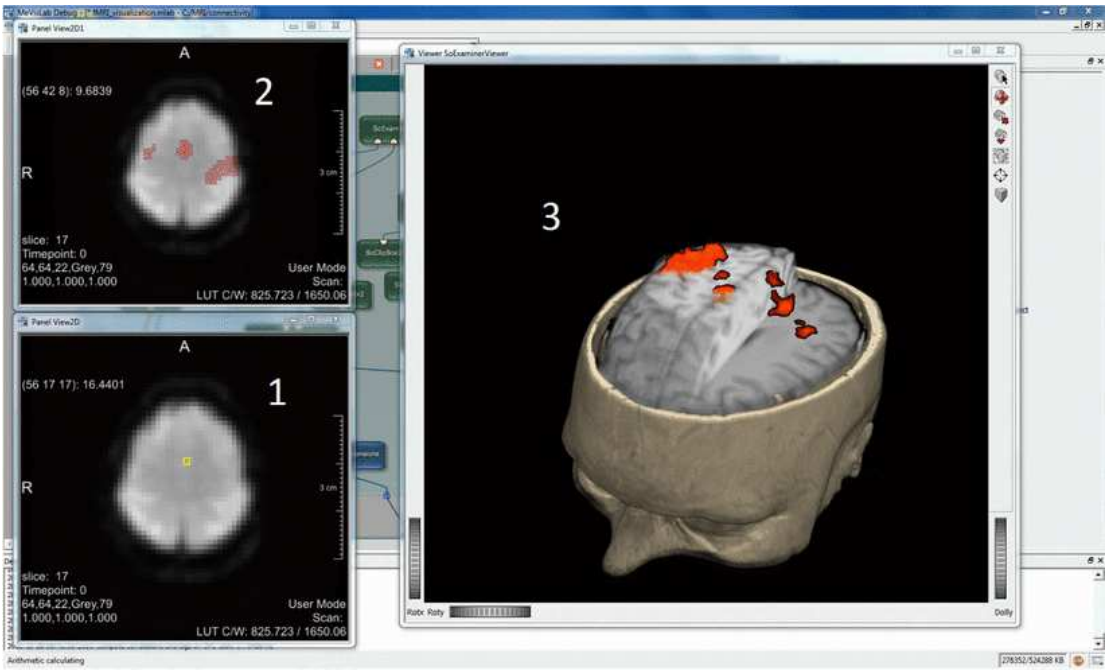


Figure 4.7: DVR of fMRI data (red) rendered together with standard MRI data (grey). Figure from: A. Eklund et al. A GPU accelerated interactive interface for exploratory functional connectivity analysis of fMRI data. 18th IEEE International Conference On Image Processing (ICIP), 2011, 1589-1592. © 2011 IEEE, used with permission.

In conclusion, reviewing the techniques for fMRI data visualisation confirms the need to display the structure, as well as the function of tissues. In particular, while fMRI maps are a powerful tool for understanding neural activity, they require context to be properly interpreted. This is similar to the material information produced by spectral CT, which may be difficult to interpret without the additional anatomical information provided by energy volumes.

4.1.3 Summary

Reviewing PET-CT and fMRI data visualisation techniques provides valuable insight into the needs of users of functional imaging modalities. In most cases, the functional information is much more useful when viewed together with the anatomical information. This is the reason for the creation of hybrid modalities, such as PET-CT, where PET provides information about the metabolic activity in the body, while CT shows exactly where this activity is taking place. The same situation can be observed in the case of fMRI, as well as SPECT-CT, which was not discussed here due to its similarity to PET-CT.

4.2 Dual-energy CT data visualisation

This section reviews the techniques currently used for visualising dual-energy CT (DECT) data. As described in section 2.2, traditional CT visualisation techniques include 2D slice display (using window and level settings to control the appearance), direct volume rendering and surface visualisation (typically through mesh extraction). Other visualisation techniques, such as maximum and minimum intensity projections (MIP and MinIP) [202, 203, 204, 205], have also been used. These are discussed in greater detail in section 7.2.

Unlike spectral CT, DECT has already been introduced into routine clinical practice, and has been demonstrated to improve diagnosis in a range of cases [45, 206]. In clinical settings, DECT data visualisation is typically performed by displaying slices of energy volumes and identified material(s) of interest. Figure 4.8 shows an example where corresponding slices of the two energy volumes acquired during the study are displayed side-by-side to emphasise the differences between them. Figure 4.9 shows a slice of an energy volume (A), and the corresponding slice of the iodine volume (B). The advantage of this technique is that no information is distorted or hidden (except by standard window/level adjustment, as discussed in section 2.2.1). This is the reason this technique is common in clinical practice [207, 208, 209, 210, 211, 212, 213].

As illustrated in Figure 4.9B, the fusion of material and energy information is not always necessary. In this case, the slice of the iodine volume provides sufficient context, and can therefore be visualised separately (although it must be noted that, in this case, the material decomposition is poor, and that many organs, such as the spine and the ribs, are erroneously classified as containing iodine). Context is

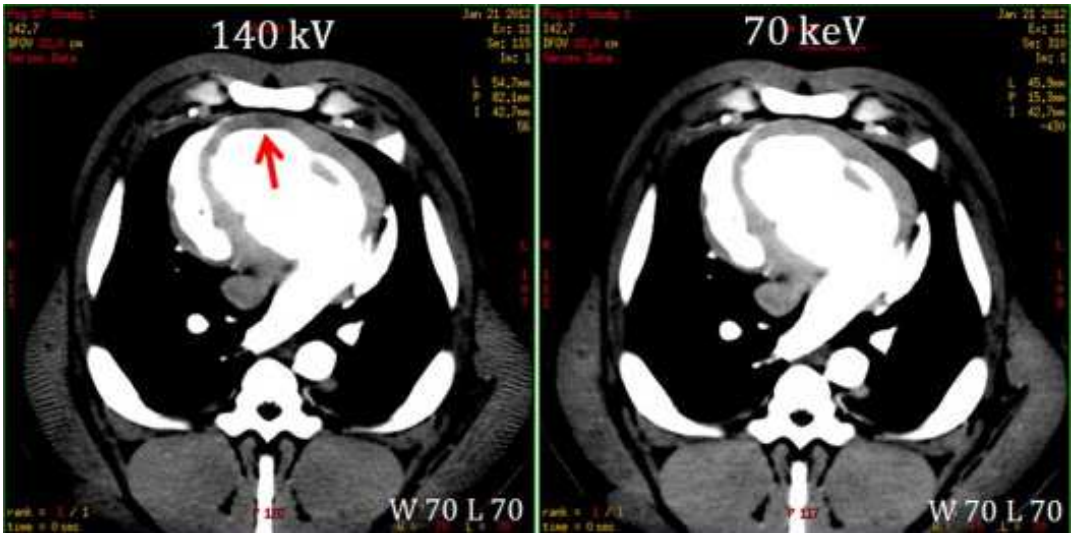


Figure 4.8: Visualisation of slices from two energy volumes (140 and 70 keV) of a dual-energy CT dataset. The area where the difference is most obvious is marked with an arrow. Figure from: A. So. Dual-energy CT and its potential use for quantitative myocardial CT perfusion. Journal of cardiovascular computed tomography, 2012, 6, 308-317. Used with permission from Elsevier B.V.

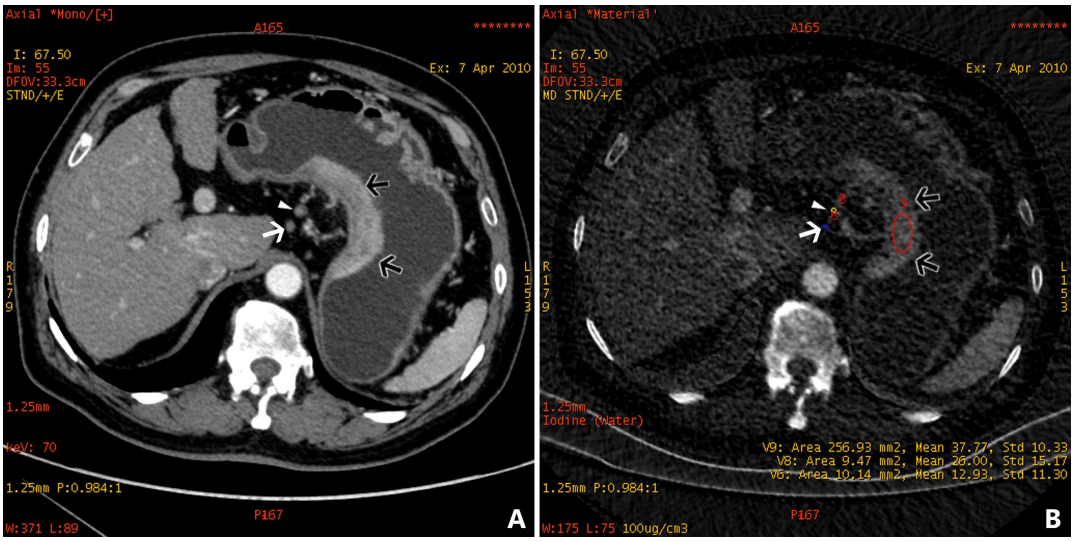


Figure 4.9: **A**: slice of the 70 keV energy volume of a clinical dual-energy CT dataset, showing the primary adenocarcinoma lesion (black arrows), and a metastatic lesion in the lymph node (white arrow). **B**: the corresponding slice of the iodine volume. Figures from: Z. Pan et al. Gastric cancer staging with dual energy spectral CT imaging. PloS one, Public Library of Science, 2013, 8 [214], used under the terms of the CC-BY 3.0 license.

Figure 4.10: DVR of uric acid deposits (shown in purple), identified using dual-energy CT. Figure from Johnson et al. Clinical image: Dual-energy computed tomographic molecular imaging of gout. Arthritis & Rheumatism, 2007, 56, 2809-2809. Reproduced with permission from Wiley Subscription Services, Inc.

required when the structures of interest are small, such as renal stones [210], uric acid deposits (a characteristic feature of gout) [42, 215], or ligaments [216]. This is illustrated in Figure 4.10, where the bones must be shown to precisely visualise the location of the uric acid deposits.

Another example is shown in Figure 4.11 (bottom row), where the quantity of fat in the liver is displayed using a "rainbow" colour scheme, while the CT image is shown in greyscale. This material overlay technique places the information provided by DECT within the anatomical context that is familiar to radiologists.

On the other hand, context is not always required in CT angiography data visualisation, where iodine is used to highlight the blood vessels. In this situation, the context (soft tissues or bones) often interferes with the visualisation of the structure of the blood vessels by occluding them. DECT allows iodine to be identified as a separate material and visualised separately, with bones being removed completely [46, 212, 217].

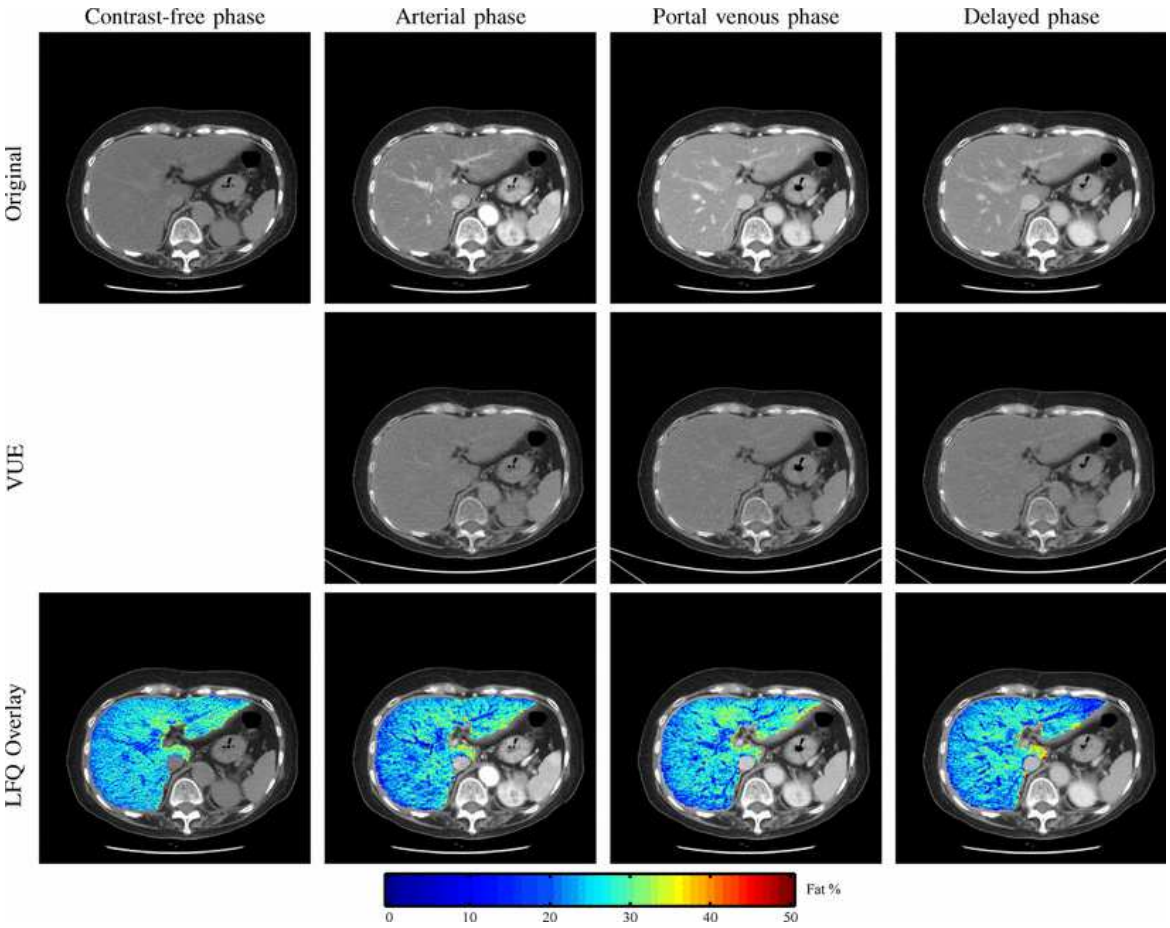


Figure 4.11: Visualisation of the quantity of fat in the liver, as identified by dual-energy CT. Fat is assigned a rainbow colour scheme, and then overlaid onto slices of an energy volume. "VUE" stands for Virtual UnEnhancement - a technique for removing contrast agents from contrast-enhanced dual-energy CT exams during visualisation. "LFQ" stands for liver fat quantification. Figure from: Mendonca et al. A Flexible Method for Multi-Material Decomposition of Dual-Energy CT Images. IEEE Transactions on Medical Imaging, 2014, 33, 99-116. © 2014 IEEE, used with permission.

4.2.1 Summary

In conclusion, it is clear that, in clinical practice, DECT data visualisation is still reliant on displaying the energy information alongside the identified materials. While improved material discrimination is the reason DECT has gained popularity in clinical practice, material information typically requires some form of context.

In addition, the problem of displaying a large number of materials (and of displaying the quantity of each material) has not been studied in detail. Usually, two materials (for example, iodine and calcium [218], or iodine and water [214]), were

identified, and quantification was limited. Spectral CT identifies more materials in a single scan; therefore, the algorithms for rendering DECT data cannot be directly applied to spectral CT data.

4.3 Spectral CT data visualisation

This section reviews the techniques previously used for visualising spectral CT data. It consists of two parts. First, it examines the research carried out by the groups not affiliated with the MARS project, and attempts to identify the methods and tools they have used for visualising spectral CT data. The second half of this section describes the work conducted by the MARS team in the area of spectral CT data visualisation before the beginning of my PhD research.

The reason these two reviews have been split is that I have in-depth knowledge of the work undertaken by the MARS team. Therefore, I can comment on both the techniques used, and the motivation for choosing them, whereas the review of the work conducted by external teams is, by necessity, restricted to commenting on the techniques used in publications.

4.3.1 Spectral CT data visualisation by other research teams

This section reviews the work on spectral CT data visualisation that was conducted by the researchers and teams not affiliated with the MARS group. It focuses on three aspects of visualisation:

1. The theory: which techniques are suitable for visualising spectral CT datasets, and why they are appropriate. This is difficult because very little research into the theory and process of spectral CT data visualisation has been published.

2. The tools: which software applications have been used for visualising spectral CT datasets and which applications have been created specifically for this purpose.

3. The results: how the results of pre-clinical studies using experimental spectral CT systems have been presented. This is an indirect approach that must be used due to the paucity of research into the theory and the tools.

This review is restricted to the studies that involved acquiring three or more energy bins. Therefore, it excludes dual-energy CT (sometimes also referred to as "spectral CT"), which has already been discussed in the previous section.

Currently, acquisition of three or more energy bins can be accomplished using:

- Commercially-available clinical CT scanners. Such systems can, at most, simultaneously acquire two energy bins using two x-ray sources and two detectors. This is currently done by some DECT scanners, such as the SOMATOM Force by Siemens Healthcare [219]. However, other techniques, such as rapid kVp switching (varying the spectrum of the x-rays emitted by the source) can be used to acquire more than two bins. In this case, the acquisition of energy bins is sequential.

- Scanners that acquire more than two energy bins simultaneously, typically using photon-counting detectors. The MARS molecular imaging system belongs to this category. This technology has not yet entered clinical use.

This review is solely concerned with visualisation, and therefore includes the studies that have employed both sequential, and simultaneous acquisition of energy bins. The remainder of this section examines individual papers, comments on the visualisation techniques used, and attempts to determine whether any research has been conducted into developing custom spectral CT data visualisation techniques.

4.3.1.1 Traditional 2D slice visualisation

This section reviews the studies that only employed 2D visualisation techniques to demonstrate the differences between energy volumes. All figures presented in these studies follow a similar pattern, with slices from multiple energy volumes being displayed side-by-side, or in a grid. An example, created using the data from the MARS system, is shown in Figure 4.12.

In 2007, Roessl and Proksa [33] conducted simulations of spectral CT imaging and presented a technique for identifying materials. Slices of phantom datasets were displayed side-by-side, and window and level settings were used to control the data ranges shown (a phantom, or an imaging phantom, is an object specially made for calibrating the scanner hardware or data processing software, or for testing a system's capabilities). The same technique was employed by Feuerlein et al. in

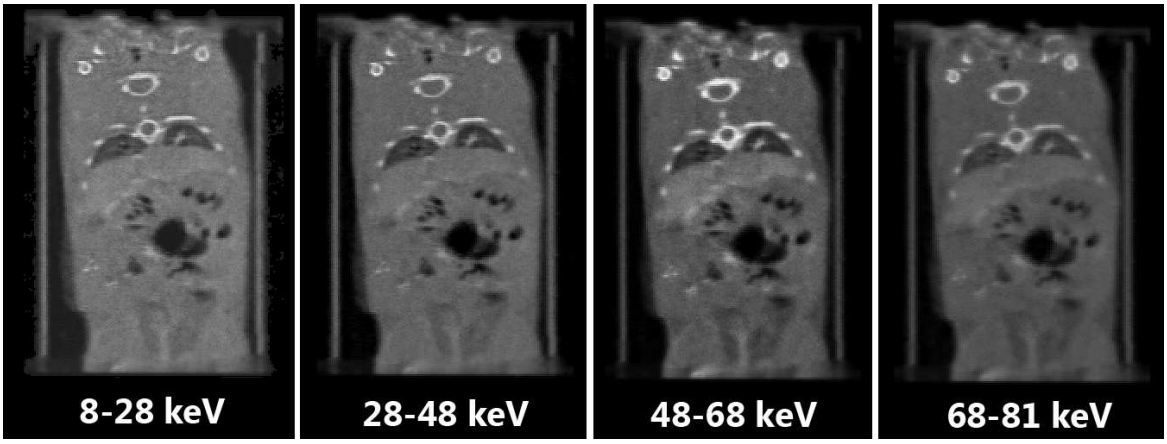


Figure 4.12: The most common visualisation technique for displaying and comparing several energy volumes, based on the review conducted in this section. A MARS spectral CT dataset (Mouse0, section 9.6.2) is used to illustrate this technique.

2008 [7], who scanned a coronary stent phantom, and Baturin et al. in 2012 [10], who displayed slices of two scanned phantoms.

In 2011, Shikhaliev and Fritz described a prototype spectral CT system that used a photon-counting detector and compared its performance to a commercial single-energy CT system [220]. In another study conducted four years later, Shikhaliev described a more advanced spectral CT system [221]. Both studies evaluated the capabilities of these systems by scanning phantoms, and in both cases visualisation was restricted to the display of 2D slices.

In 2010, Boll et al. used an experimental photon-counting spectral CT system provided by Philips Research to scan a cystic renal lesion phantom [12]. Slices from five energy bins were visualised next to each other to show the difference in the attenuation profiles of saline solution, blood and iodine solution. Analysis was performed using an application called Brilliance Workspace, developed by Philips Healthcare.

In 2011, Leng et al. investigated the use of multiple energy bins for reducing the noise present in spectral CT datasets [157]. Denoising was incorporated into a reconstruction algorithm, which was then successfully applied to real data acquired by a commercial dual-energy CT system. Slices of original and denoised energy volumes were shown side-by-side to visualise the improvement.

In all six cases described above, scanning was largely restricted to phantoms, which are the easiest type of object to visualise, as their dimensions and structure,

as well as the locations of features of interest, are known in advance. For example, most phantoms scanned during these studies were tubes containing aqueous solutions of different concentrations of materials. Therefore, the geometry was simple, and there was little difference between the slices.

The basic visualisation techniques employed by the authors of these papers were sufficient to show the difference between the energy volumes. Due to the simple geometry of the scanned objects, there was no need to show the relationship between the various structures inside them. If that was necessary, then 2D slice visualisation may not have been sufficient, as it only shows a single slice at a time. 3D techniques may be better for visualising objects with a more complex internal structure, such as excised tissue samples, small animal specimen, or human patients. For example, volume rendering has been demonstrated to be useful for visualising CT datasets in clinical practice [202, 222, 223].

In conclusion, the six studies reviewed in this section did not show evidence of any custom spectral CT data visualisation technique being used. However, these studies also demonstrate that traditional 2D visualisation using a greyscale colour scheme (controlled by window and level settings) is currently considered an essential technique for displaying the differences between energy volumes. Therefore, any spectral CT data visualisation software should implement this visualisation style, along with any novel approaches. This includes the software that I have developed over the course of my PhD research, as described in section 5.6.

4.3.1.2 3D visualisation

3D rendering of spectral CT data has been used by Iwanczyk et al. [51], who employ both 2D, and 3D visualisation techniques to present the results of human carotid artery scans. As usual, slices of different energy volumes are displayed side-by-side. In addition, a volume rendering and a maximum intensity projection (MIP) rendering are included.

However, it is unclear which energy volume has been visualised and which tools have been used. The 3D rendering appears to show a carotid artery highlighted by iodine, which was used as the contrast agent. Overall, little context is provided, and it appears that DVR and MIP are only used to demonstrate the image quality obtained by the spectral CT system used by Iwanczyk et al.

Two papers by Pan et al. [224, 225], published in 2010 and 2012, respectively,

also use 3D visualisation of spectral CT data. Both papers describe the development of novel nanoparticles that may be used as biomarkers and detected by spectral CT.

The first paper describes the development of a novel bismuth-based nanoparticle targeted at thrombus imaging. Small animal testing confirms that an experimental spectral CT system from Philips Research can identify this nanoparticle. Interestingly, a 3D visualisation of bones and a thrombus containing the nanoparticles is shown. This image is reproduced in Figure 4.13. Neither the software, nor the exact rendering techniques were specified.

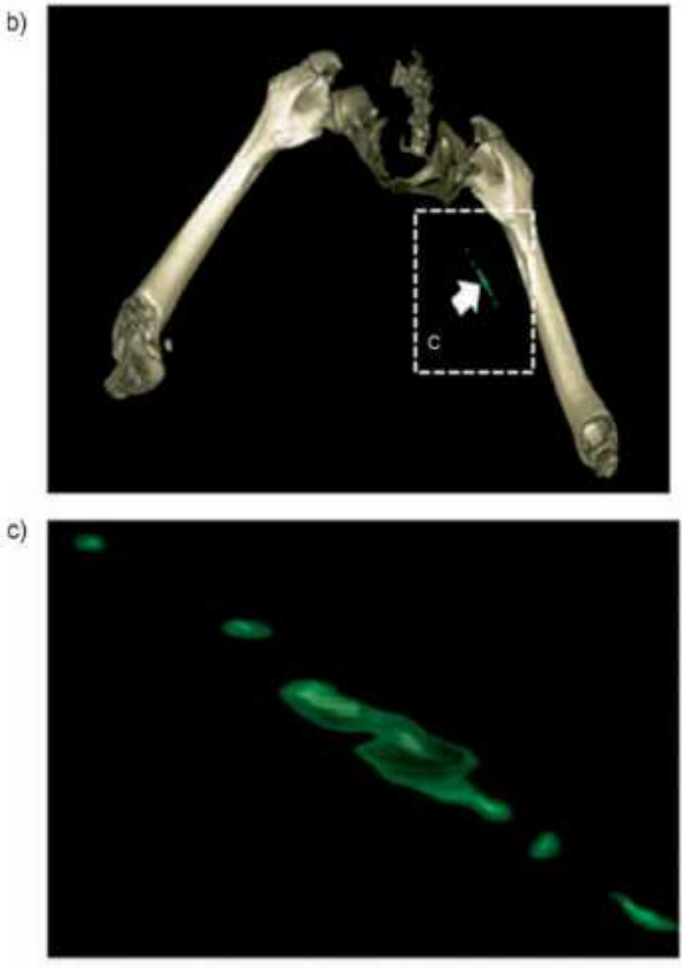


Figure 4.13: Visualisation of bones and a region containing a contrast agent. Figure from: Pan et al. "Computed Tomography in Color: NanoK-Enhanced Spectral CT Molecular Imaging" Angewandte Chemie, WILEY-VCH Verlag, 2010, 122, 9829-9833. ©2010 WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim. Reproduced with permission.

The second paper focuses on the development of an ytterbium-based contrast agent for spectral CT imaging. Testing is conducted using phantoms and a small animal (mouse) model. The visualisation of the results is similar to the first paper. The quantification of ytterbium in the scanned phantom is shown by fusing a slice of an energy volume (used as a base and displayed using a greyscale colour scheme) with a slice of the ytterbium volume, where the concentration of this material is shown as a colour gradient. In addition, a 3D volume rendering of a skeleton of a mouse is provided. A small region corresponding to the location where the contrast agent has accumulated is shown in a different colour.

However, in both papers by Pan et al., no information about the rendering algorithms or visualisation software is provided. In both cases, the anatomical context is provided by the bones (which may have been extracted from an energy volume, or identified using some form of MD), while the region of interest is shown in a contrasting colour. This is similar to the existing DECT data visualisation.

In conclusion, very few of the reviewed papers employ any sort of 3D visualisation. If it was used, then few details about the rendering techniques were provided. Overall, no evidence of research into the requirements or techniques for 3D visualisation of spectral CT data could be found.

This thesis develops novel 3D rendering techniques and tools for clearly displaying regions of interest during 3D visualisation of spectral CT data. These are discussed in section 5.3 and Chapter 7, respectively.

4.3.1.3 Human pre-clinical trials

Lv et al. configured a commercial single-source CT system to use rapid kVp switching and acquired a series of energy bins [8]. Their study investigated the diagnostic accuracy of using different energy bins to detect hepatocellular carcinoma (a type of liver cancer). The accuracy was evaluated by experienced radiologists, who found that the optimal energy range for diagnosis was 40-70 keV, as it allowed them to clearly see the features most commonly associated with this type of cancer.

This study is valuable because it clearly demonstrates the benefit of measuring a specific energy range targeted at a particular problem. However, from the point of view of visualisation, no novel techniques have been used: the radiologists performing the diagnosis treated each energy volume as a separate single-energy

CT dataset. Standard 2D visualisation of axial slices was used for diagnosis; slices of different energy bins were visualised separately (that is, not combined in any way).

The study by Iwanczyk et al. [51], mentioned earlier in this review, also conducted a human clinical trial. However, the trial was conducted to demonstrate the feasibility of clinical spectral CT imaging with the system described in the paper; investigating the clinical requirements for spectral CT data visualisation was not a priority. Instead, slices of different energy volumes were visualised using standard 2D techniques.

Overall, few clinical trials have been conducted, and the visualisation techniques used for diagnosis have not deviated from standard approaches. No custom tools have been described.

4.3.1.4 Use of colour coding

Two of the reviewed studies are notable for employing colour coding to differentiate between several identified materials. In 2008, Schlomka et al. [48] used an experimental spectral CT system designed by Philips Healthcare to scan a phantom with varying concentrations of iodine and gadolinium. Slices of energy volumes were displayed side-by-side to clearly visualise the differences. Material decomposition was performed using the method described in the earlier paper by Roessl and Proksa [33]. Identified materials were assigned distinct colours and overlaid on top of a slice of an energy volume, which was shown in greyscale. A recreation using the data from the MARS system is shown in Figure 4.14.

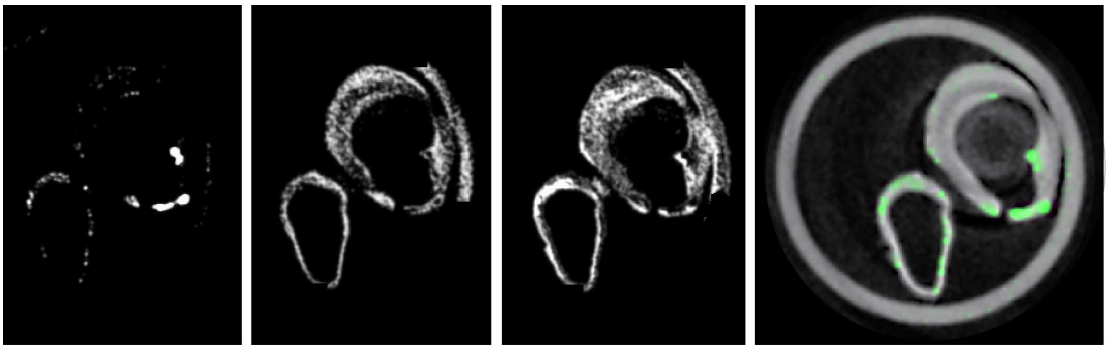


Figure 4.14: The most common visualisation technique for displaying different material channels and for overlaying a slice of a material volume onto a slice of an energy volume. A MARS spectral CT dataset (Plaque72, section 9.6.2) is used to illustrate this technique.

In 2010, Cormode et al. [5] used the same prototype system to scan a phantom of an artery, as well as mice injected with gold nanoparticles. The same method was also used for material decomposition, which allowed the gold and the iodine to be separated. Slices of both materials were visualised individually, as well as in a single combined 2D image, where colour coding was used to separate the material channels.

However, these papers do not focus on any aspect of the process of spectral CT data visualisation. For example, the use of colour is clearly a useful technique for visually separating different channels of a spectral CT dataset. Schlomka et al. and Cormode et al. employ this technique, but do not discuss how it may be extended, and which problems may be associated with it.

For example, the paper by Cormode et al. shows four colour-coded materials in a single image. However, the blending between the colours assigned to each material distorts the colour gradient for each one. Another image shows an identified material (gold) overlaid onto a slice of an energy volume. Again, blending with the greyscale colour scheme assigned to the energy volume distorts the colour assigned to the gold channel. An example of this problem, recreated using the data from the MARS system, is shown in Figure 4.15. The colour assigned to a particular material (in this case, iodine), is distorted by blending with the original energy volume.

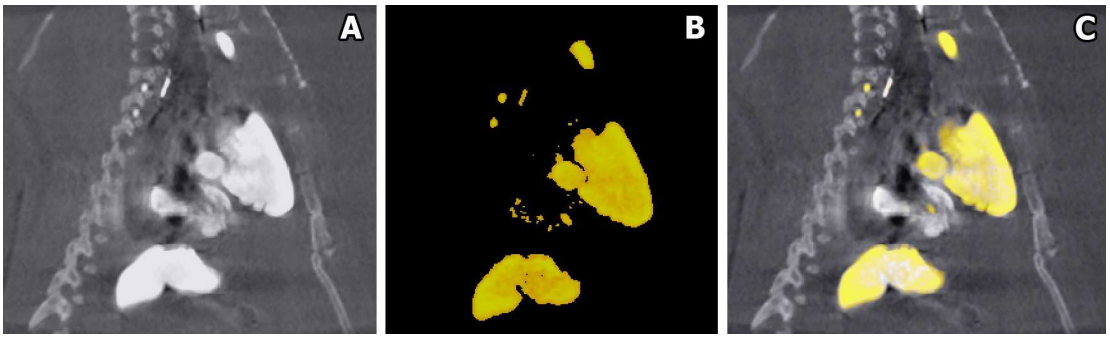


Figure 4.15: Colour distortion during the fusion of data from an energy volume and a material volume, using the Mouse12dataset (section 9.6.1), as an example. **A**: energy volume only. **B**: material volume only. **C**: fusion of energy and material data. The colour gradient assigned to the material volume is distorted.

Overall, the use of colour is clearly a promising technique for differentiating between materials. However, attention has not been paid to the implementation

of colour blending algorithms, and no tools for performing data fusion have been described in the reviewed papers.

This research addresses this problem by:

1. Creating data fusion algorithms capable of either blending all visible channels of a spectral CT dataset, or displaying the colour for a certain channel without blending. These algorithms are described in sections 5.6 and 5.3, respectively.

2. Designing a novel transfer function editor for assigning colour to both energy, and material volumes of a spectral CT dataset. This tool is described in Chapter 6.

4.3.1.5 Summary

This review could not find any research dedicated to the theory, the process or the algorithms for spectral CT data visualisation. The most common visualisation technique used by the other research teams appears to be side-by-side display of slices of different energy volumes in order to show the differences between them. Appearance is typically controlled using window and level settings.

The studies that applied some form of material decomposition usually employed the same technique to display the slices of individual material volumes. It must be noted that the term "material volume" is not used in these papers. However, they display slices of volumetric datasets that represent the concentration of materials. The convention used in this thesis refers to these datasets as "material volumes" (section 3.2.2).

Sometimes, several materials were combined in a single image and colour was used to differentiate between them. Material information was also overlaid onto energy information, in which case materials were assigned colours, and slices of energy volumes were displayed in greyscale.

The reviewed papers used visualisation to present the results of pre-clinical studies, and did not focus on the specific software, or the tools used to generate the images. However, it is likely that general-purpose image processing applications such as MATLAB or ImageJ, or perhaps a DICOM viewer, have been used. Regardless of the tools used, no image in the reviewed papers suggests that any

custom techniques have been developed specifically for visualising spectral CT data.

4.3.2 Spectral CT data visualisation by the MARS team

This section discusses the visualisation methods and tools used by the MARS team before the beginning of my PhD research. It describes the problems that were encountered, the reasoning behind some of the decisions regarding the choice of visualisation techniques, and the deficiencies of the tools available at that point in time.

The datasets processed by the MARS toolchain have been visualised using both general-purpose visualisation software, and custom applications. Several applications have been used depending on the task and the technical ability of the person conducting the study. The software ranged from standard scientific image processing tools such as MATLAB [153] and ImageJ [67] to MARSCTExplorer, an application specifically created for visualising spectral CT datasets.

The visualisation of both energy and material information has been examined. Material decomposition, as described in section 3.2.2, is a relatively late development, as current MD algorithms have only been available since early 2013 [36]. Previously, the most common approach has involved using ImageJ to display slices of different energy volumes separately, as shown in Figure 4.16. Examples include the papers by Walsh et al. [58], Zainon et al. [57], Butler et al. [226], Butzer [54], and Ronaldson et al. [227].

As explained in section 4.3.1.1, the same approach was used by other research teams. It is conceptually simple, and does not require designing custom software or rendering algorithms. An energy volume is treated as a conventional CT dataset, which means that the features offered by general-purpose software such as ImageJ, or any DICOM viewer, are sufficient.

However, it quickly became clear that displaying energy volumes separately limited the amount of information conveyed by visualisation. Instead, the information contained in energy volumes needed to be processed and combined in some manner in order to extract useful features or materials.

Two methods have been investigated: combining (also known as intermixing) energy volumes during visualisation, and material decomposition using principal component analysis (PCA). These methods have been studied in 2009-2012, before

the introduction of the MARS MD algorithms in use today.

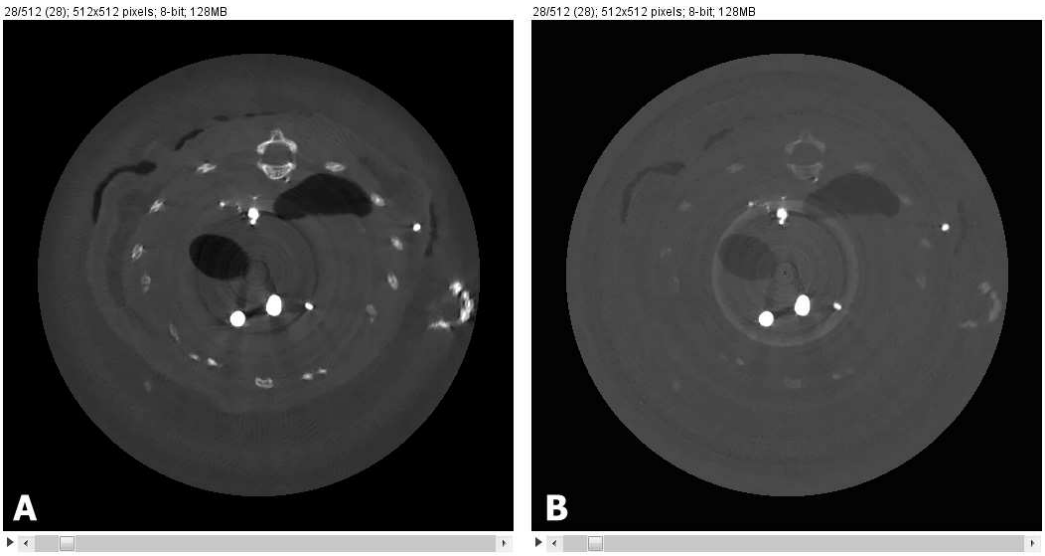


Figure 4.16: Visualisation of a slice of the 15-80 keV energy volume (**A**) and the corresponding slice from the 35-80 keV energy volume (**B**) of the Mouse12 dataset (section 9.6.1) with ImageJ.

4.3.2.1 Real-time intermixing of energy volumes

This approach relies on the user combining data from different energy volumes in order to extract useful information. The combination is carried out interactively during visualisation. This, in theory, allows the user to rapidly experiment with different combinations and identify promising ones.

Extraction of features during intermixing is theoretically possible because each energy volume in a spectral CT dataset is a slightly different representation of the scanned object. Therefore, it should be possible to accentuate these differences, which may correspond to certain structures or materials of interest.

Basic intermixing can be performed with ImageJ, as shown in Figure 4.17. Each volume can be assigned a certain pre-defined colour, but there is no fine control over the colour gradient, or the data ranges shown. The structures of interest can be better highlighted by performing volume data processing (such as scaling, addition, or subtraction) prior to intermixing. However, it cannot be considered a permanent solution for energy volume intermixing, as the process is highly manual and requires an intimate knowledge of ImageJ's toolset.

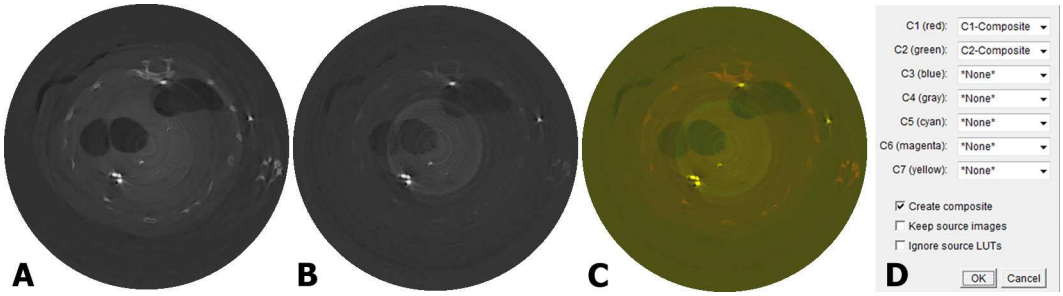


Figure 4.17: Using ImageJ to fuse slices of two energy volumes of the Mouse12 dataset (section 9.6.1). **A**: the 15-80 keV energy volume. **B**: the 35-80 keV energy volume. **C**: fused image, with the 15-80 keV volume assigned to the red channel and the 35-80 keV volume assigned to the green channel. **D**: ImageJ's "Merge Channels" GUI window.

The first examination of MARS spectral CT data visualisation was carried out by Niels de Ruiter in 2009-2011 [129]. De Ruiter identified the stages at which energy volumes may be combined and developed a software framework for performing intermixing, known as MARSCTExplorer. In MARSCTExplorer, energy volume intermixing was based on arithmetic operations (addition, subtraction, multiplication or division), binary logic operations (such as AND, OR, XOR), and vector and matrix multiplication. A basic transfer function editor was also included.

MARSCTExplorer allowed the user to intermix energy volumes at different stages of the visualisation pipeline (identified by Cai et al. [228]), as shown in Figure 4.18. Figure 4.19 shows the GUI of MARSCTExplorer, while Figure 4.20 provides an example of the visual quality achieved its DVR algorithm.

MARSCTExplorer was used for visualising the results of several studies by the MARS group [6,57,229]. However, its utility was limited because it only supported visualisation using maximum intensity projection, isosurface rendering, or DVR, contained no measurement or segmentation tools, and implemented complicated data intermixing concepts.

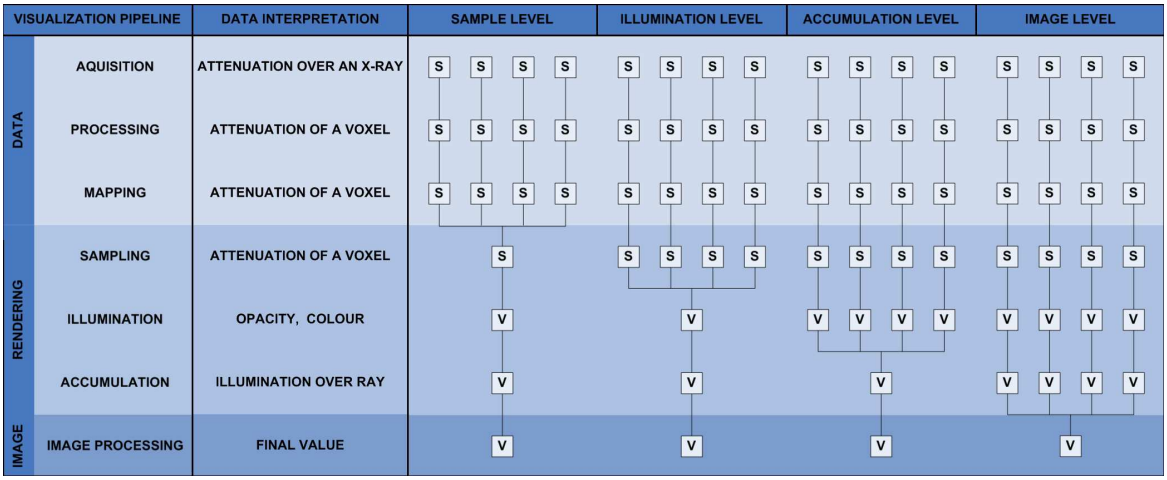


Figure 4.18: Intermixing as it applies to spectral CT data visualisation. The visualisation pipeline consists of several stages. Energy volume data can be combined during the sampling, illumination or accumulation stages. Image courtesy of Niels de Ruiter.

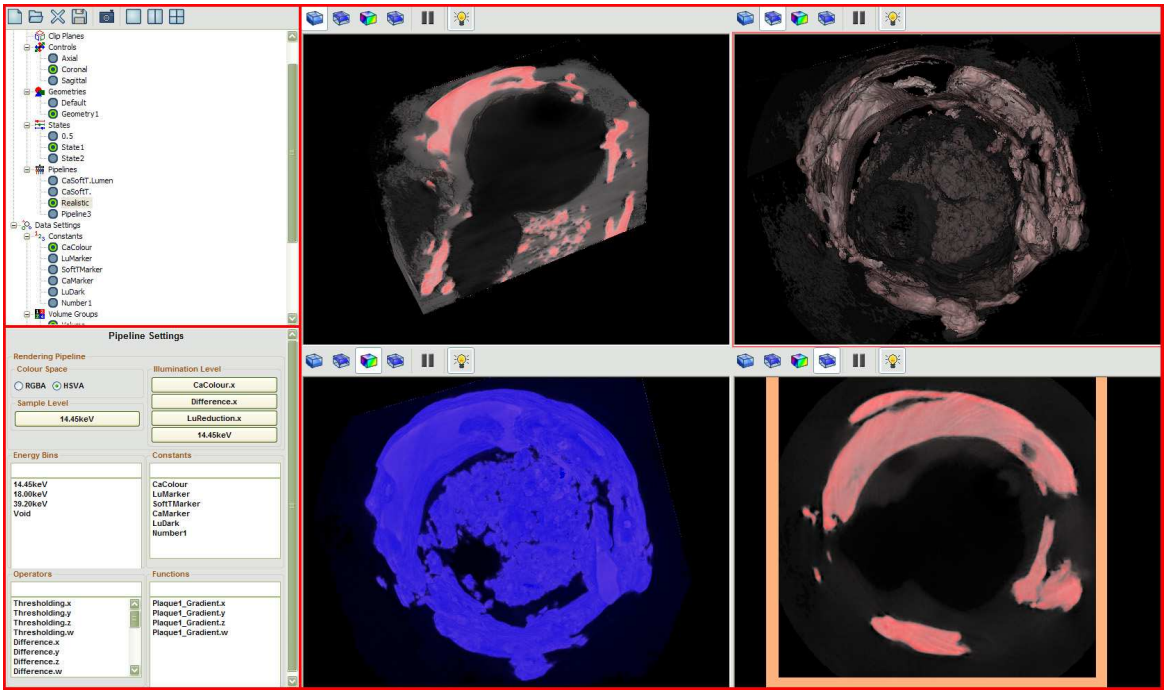


Figure 4.19: The GUI of MARSCTExplorer. The four views show (clockwise from top left): DVR; isosurface rendering; maximum intensity projection; DVR with the camera and the clipping plane set to approximate a 2D slice view. Image courtesy of Niels de Ruiter.

The feedback gathered from users of MARSCTExplorer pointed towards the need to simplify the user interfaces used for colour assignment and data fusion. For example, the use of window and level settings to adjust the appearance of 2D slices was well-known by the pre-clinical researchers associated with the MARS team. However, no pre-clinical researcher was familiar with the process of designing transfer functions for DVR visualisation. This was a very important realisation that has influenced the design of the GUIs presented in this thesis.

4.3.2.2 Material decomposition with PCA

Early attempts at extracting material information have focused on using PCA. This approach has identified materials such as calcium, iodine and barium [6, 53, 55], but was not used extensively. In addition to real-time intermixing of energy volumes, MARSCTExplorer was also capable of calculating and applying a PCA covariance matrix to separate materials, as shown in Figure 4.20.

Figure 4.20: The Mouse12 dataset (section 9.6.1) visualised with MARSCTExplorer. Materials have been identified with PCA. Calcium is shown as orange, barium as blue and iodine as green. Image courtesy of Niels de Ruiter.

4.3.2.3 *Visualisation of compressed spectral CT datasets*

My first contribution to spectral CT data visualisation has been made in 2010-2012, as part of my MSc research project. MARS spectral CT datasets are significantly larger than conventional CT datasets because they contain several channels (energy or material volumes). Meanwhile, volume rendering algorithms are usually executed on GPUs, which contain a limited amount of memory to store volume data. This concern has driven the development of compression techniques for spectral CT data.

My research focused on rendering directly from a compressed form of a spectral CT dataset stored inside the memory of a GPU. I have implemented two different compression algorithms, with compression ratios ranging from 2:1 in the worst case, to 64:3 in the best case. Figure 4.21 shows two examples of images rendered directly from a compressed spectral CT dataset. The software created over the course of my MSc research was written in the CUDA GPGPU language, and was called MARSCUDAVR.

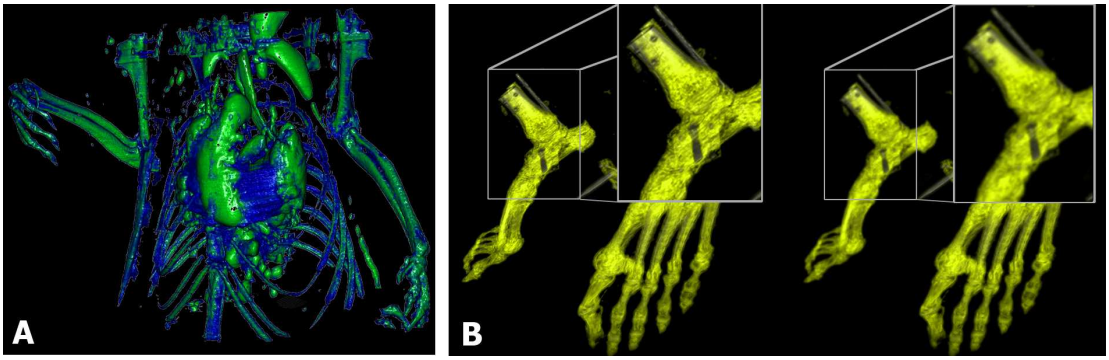


Figure 4.21: DVR of two datasets using MARSCUDAVR. **A**: compressed Mouse12 dataset (section 9.6.1). **B**: compressed Vakhum Mummy conventional CT dataset [230], showing depth-of-field effects added in post-processing.

This project focused on compression alone, and GUI design, intermixing algorithms, and other aspects of spectral CT visualisation were not studied. 2D visualisation techniques and segmentation and measurement tools were not implemented.

Nevertheless, my research has shown that modern GPGPU languages such as CUDA (section 2.3) are well-suited for implementing 3D rendering algorithms for spectral CT data visualisation. These languages provide a number of advantages

over shading languages, such as full read/write access to any correctly declared and allocated array in GPU memory. This can be used to add post-processing effects such as edge detection or depth-of-field simulation (Figure 4.21B) to the basic rendering algorithm.

Finally, this project has demonstrated the importance of optimising DVR algorithms for rendering spectral CT datasets. It has shown that using a GPU-based octree to perform empty-space skipping results in a 200-400% performance increase. This knowledge has been valuable for designing the volume rendering algorithms described in this thesis.

4.3.2.4 Summary

The MARS team has investigated the process of spectral CT data visualisation and the use of various visualisation techniques to present the results of pre-clinical research. In particular:

- Results of pre-clinical studies have usually been presented using 2D techniques. Slices from several energy volumes have been displayed side-by-side in order to clearly show the difference. 3D rendering has been used in rare cases, usually to supplement 2D visualisation.

- DVR of multiple volumes of a spectral CT dataset has been shown to be possible on consumer desktop PC hardware.

- The design of custom tools for spectral CT data visualisation was examined briefly, but was not an important research topic.

- The graphical user interface for performing data fusion and assigning colour to energy or material data was complex and unfamiliar to pre-clinical researchers and radiologists associated with the MARS team.

In conclusion, before the beginning of this research project, MARS users mostly used generic visualisation and image processing applications. Custom visualisation software existed, but was not often been used, as it only supported 3D techniques such as DVR, as opposed to the familiar 2D slice visualisation techniques.

4.3.3 Summary

This section has shown that the process of visualising spectral CT data has not been examined in detail. Aside from the work by the MARS group, specialised tools for visualising spectral CT datasets do not appear to have been developed.

The vast majority of reviewed studies have used 2D slice visualisation. Different energy volumes have not been combined during visualisation. Materials, if identified, have been shown separately, or have been overlaid on top of a slice of an energy volume. 3D rendering techniques, such as DVR, have been used sparingly, and have nearly always been paired with 2D visualisation.

The MARS team has used the techniques described above, but has also developed custom algorithms and software for intermixing energy volumes during interactive visualisation. However, special tools for 2D visualisation have not been created. Instead, general-purpose image processing software has been used for displaying slices and performing measurements.

In conclusion, the only known research into the process and the tools for visualising spectral CT datasets has been published by the MARS team. Other groups appear to use generic visualisation software, usually treating individual energy or material channels as separate volumetric datasets.

4.4 Requirements for visualisation of MARS energy and material data

This section discusses the requirements for visualising the energy and material volumes produced by the MARS toolchain. However, the spectral CT data structures used by the MARS project have no features directly tying them to the MARS scanner hardware [59]. That is, all data is stored in the standard DICOM format, the energy volumes are stored in units of linear attenuation, while material volumes are stored in units of mass concentration.

This section attempts to summarise existing research and establish a set of features that leads to effective visualisation of results of pre-clinical studies. These requirements are based on the literature review conducted in the previous section, on the experience of the MARS team, and on related research into the tools and requirements for medical visualisation in general.

4.4.1 Channel selection and data fusion

Spectral CT measures the attenuation of photons across several energy ranges of the x-ray spectrum. Reconstruction creates energy volumes, which may be studied without further post-processing [9, 13, 33, 51]. In addition, material information may also be extracted in some cases [4, 5, 6, 48].

The common feature is the presence of multiple independent volumetric datasets, or channels. Currently, a channel can be defined as a single energy or material volume, as explained in section 3.2. Each channel can be visualised or measured separately.

Therefore, the first and most essential requirement for spectral CT data visualisation is control over which channels are displayed and over the appearance of each channel. Previously, this has been achieved by visualising slices of different energy or material volumes separately, as described in section 4.3. Usually, slices of several energy volumes have been displayed side-by-side, or in a grid. In this situation, the user has full control over the appearance of each energy volume, as the visualisation parameters (for example, window and level settings) are completely independent.

However, if several channels are combined during visualisation, then the user must be given the tools for selecting which channels to display and the GUI elements for performing data fusion and making channels appear distinct. These tools must allow the user to individually adjust the appearance of each channel.

The fusion of different channels of a spectral CT dataset is not a novel idea. Section 4.3 has described the use of this technique to display the results of pre-clinical studies. However, only one identified study (by de Ruiter [129], discussed in detail in section 4.3.2.1) considered the tools and GUI elements used to set the fusion parameters. Therefore, further research is required.

This thesis attempts to address this problem by presenting a custom tool for facilitating the fusion of material channels of MARS spectral CT datasets. This tool is described in Chapter 6.

4.4.2 2D visualisation

Reviewing the techniques used for visualising spectral CT data uncovers a gap in the existing research. 2D slice visualisation has been the preferred method of displaying the results of pre-clinical studies. However, no custom algorithms or

tools for 2D visualisation of spectral CT data have been described in literature. This suggests that researchers have only focused on the end result of visualisation, and not on the process. For example, the algorithms for fusing the slices of multiple channels, the tools for assigning colour to these channels, and the advantages and limitations of this technique have not been investigated.

This is a serious omission, as the use of 2D visualisation is widespread in clinical practice. For example, several studies that evaluate the capabilities of DICOM viewers from a radiologist's point of view [80, 81, 231] emphasise the importance of a complete toolset for 2D slice visualisation and measurement. In contrast, 3D visualisation is not mentioned as being an important part of the radiologist's workflow.

The needs of medical professionals (the likely future users of spectral CT) and scientists (its current users) differ, so reviewing medical literature alone is insufficient. However, based on the review in section 4.3, we can conclude that 2D slice visualisation is also the most popular technique used by the scientists currently studying spectral CT. It is also the most common method of visualising MARS data.

Therefore, 2D visualisation will likely continue to be used by both the current users of prototype spectral CT systems (including MARS), and the vast majority of future clinical users of spectral CT. This means that the tools and GUI elements for performing slice visualisation should be customised to take advantage of the additional information provided by spectral CT. In particular, 2D fusion of multiple volumes, analogous to that already achieved using DVR, should be investigated. In addition, the techniques developed must be able to show the context along with the material, or materials of interest. This can be achieved, for example, by overlaying material information onto a slice of an energy volune. This work describes the algorithm for 2D spectral CT data fusion in section 5.6.

4.4.3 3D visualisation

This section discusses three issues regarding the performance and visual quality of 3D rendering algorithms for spectral CT data visualisation. These are:

1. Is 3D visualisation useful?

2. What image quality should 3D visualisation provide?

3. What should the performance of 3D visualisation be?

The first issue concerns the need for 3D visualisation. As mentioned in section 4.4.2, it is not currently considered essential to a radiologist's workflow. Nevertheless, it has been demonstrated to be useful in clinical practice.

Fishman et al. [223] advocate the use of DVR and maximum intensity projection (MIP) for CT angiography. This view is supported by Calhoun et al. [202], who also suggest other possible uses, such as diagnosing aneurysms, tumours and artery stenosis. Perandini et al. [222] conduct a brief review of clinical applications of volume rendering. This review identifies the applications mentioned above, along with others, such as virtual endoscopy, virtual colonoscopy and maxillofacial surgery planning.

The experience of the MARS team with 3D visualisation of spectral CT datasets is limited to pre-clinical research, but it confirms the observations of clinicians. 3D rendering techniques, such as DVR, help the user visualise the position of various structures in a dataset and the relationship between these structures. However, such techniques are usually insufficient to fully explore the dataset due to occlusion and the difficulty of designing transfer functions and performing measurements in 3D space.

Therefore, we can conclude that 3D visualisation is potentially useful, but that it cannot yet replace traditional 2D techniques. Instead, it should be integrated with 2D slice visualisation and the user's actions should be synchronised, if possible. For example, the user may be allowed to select features or place clipping planes in 3D, while the individual slices that contain these features can be shown alongside. This technique is illustrated in Figure 4.22. Another option is the synchronisation of annotations placed in 2D and 3D views, as shown in Figure 4.23.

The second question concerns image quality and photorealism. The algorithms previously used for rendering MARS spectral CT datasets did not generate photorealistic images, although the images were comprehensive enough to identify numerous structures and regions of interest.

The aim of medical visualisation is to help the user make a diagnosis; the techniques used to achieve this are of secondary importance. Thus, photorealism may not always be desirable. In some cases, non-photorealistic rendering techniques may convey more information than standard methods that aim for photorealistic

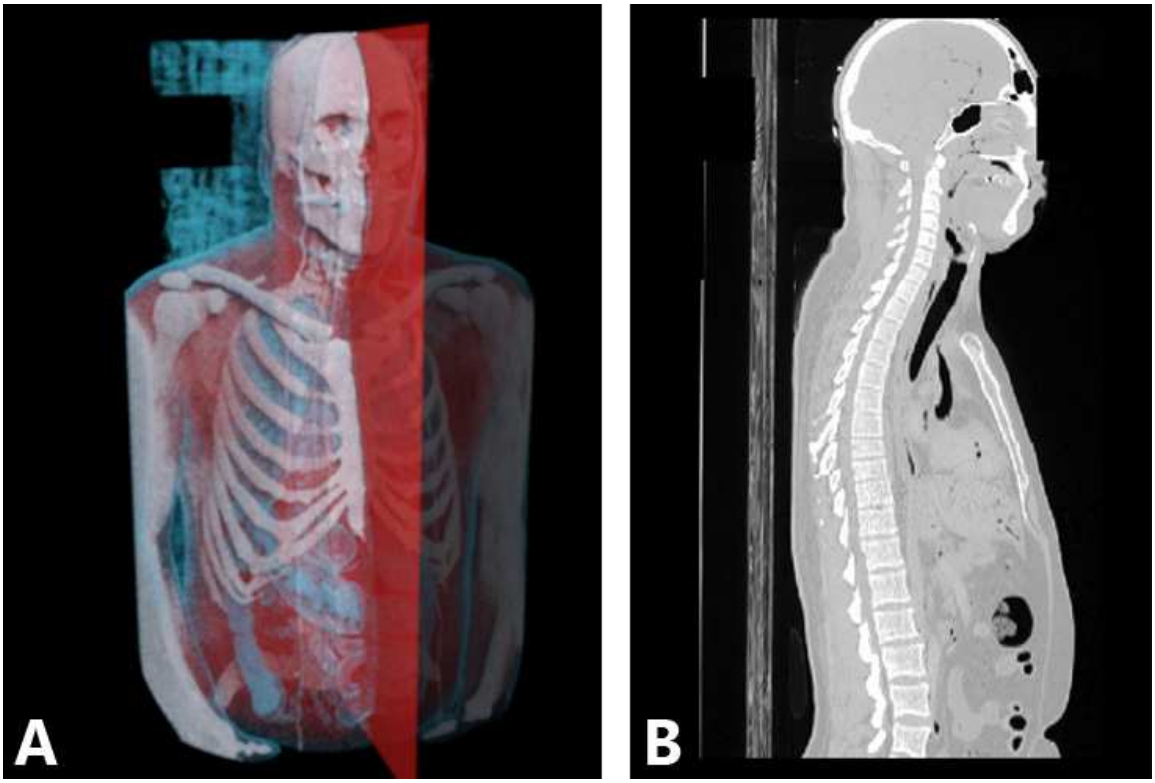


Figure 4.22: A possible technique for synchronising 2D and 3D visualisation. The user moves an axis-aligned plane in the 3D view (**A**). The slice that corresponds to the position of the plane is displayed using standard window and level settings (**B**).

image quality. This explains the popularity of maximum intensity projection (further discussed in section 7.2.1), which does not render photorealistic images, but removes occluding soft tissues [202, 203, 223, 232]. Other options include emphasising contours [233], or the internal features of the dataset [123].

On the other hand, high-quality simulations of light transfer through the dataset, realistic shadows, scattering and reflections provide depth cues and help the user visualise the position of structures and the details of surfaces [60, 234].

Therefore, there is no single correct answer to this question. Both photorealistic, and NPR techniques have been successfully used for visualising medical datasets. Ideally, a 3D rendering algorithm should be flexible so that it can be extended or modified to allow for various techniques to be tested.

The final question concerns the performance of 3D visualisation. Here, a definitive answer can be given. Visualisation must be interactive, regardless of the specifics of the algorithm. This means that the user should be able to see the

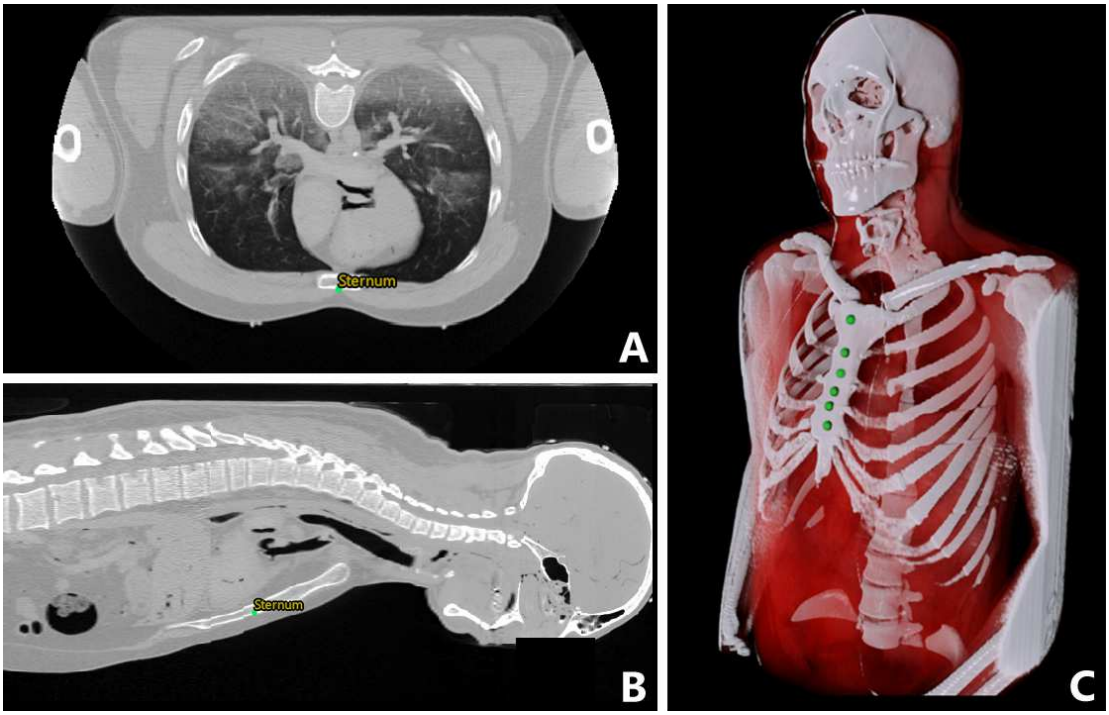


Figure 4.23: A possible technique for synchronising the display of annotations during 2D and 3D visualisation. The annotation points can be overlaid on top of a slice (**A**,**B**), or rendered inside a volume (**C**).

results of input immediately. Usually, frame rates of above 15 frames per second are considered interactive [235, 236].

Early work on 3D rendering of spectral CT datasets has confirmed the feasibility of achieving interactive performance on commodity GPU hardware [74, 129]. That work has been conducted several years ago and GPU hardware has since improved dramatically. Therefore, an algorithm that provides both high-quality visualisation, and acceptable performance on commodity hardware is certainly a realistic expectation.

4.4.4 Occlusion and visibility of features

The visibility of features and regions of interest during 3D volume visualisation is a serious problem [100]. On the one hand, occlusion is an important depth cue to the human visual system [237], and depth perception has been identified as a major advantage of 3D rendering techniques over traditional 2D visualisation for clinical diagnosis [202]. On the other hand, occlusion may obstruct some features

of interest.

In a reconstructed CT dataset, nearly all structures of interest lie inside the object. However, this is also where the effects of occlusion are most significant. The same is true for spectral CT datasets. For example, consider a DVR of three material channels shown in Figure 4.24. All materials are set as being opaque, so it is natural to expect the structures close to the camera to occlude the structures located further away, as it imitates the appearance of real materials. However, these settings also cause occlusion of important internal features.

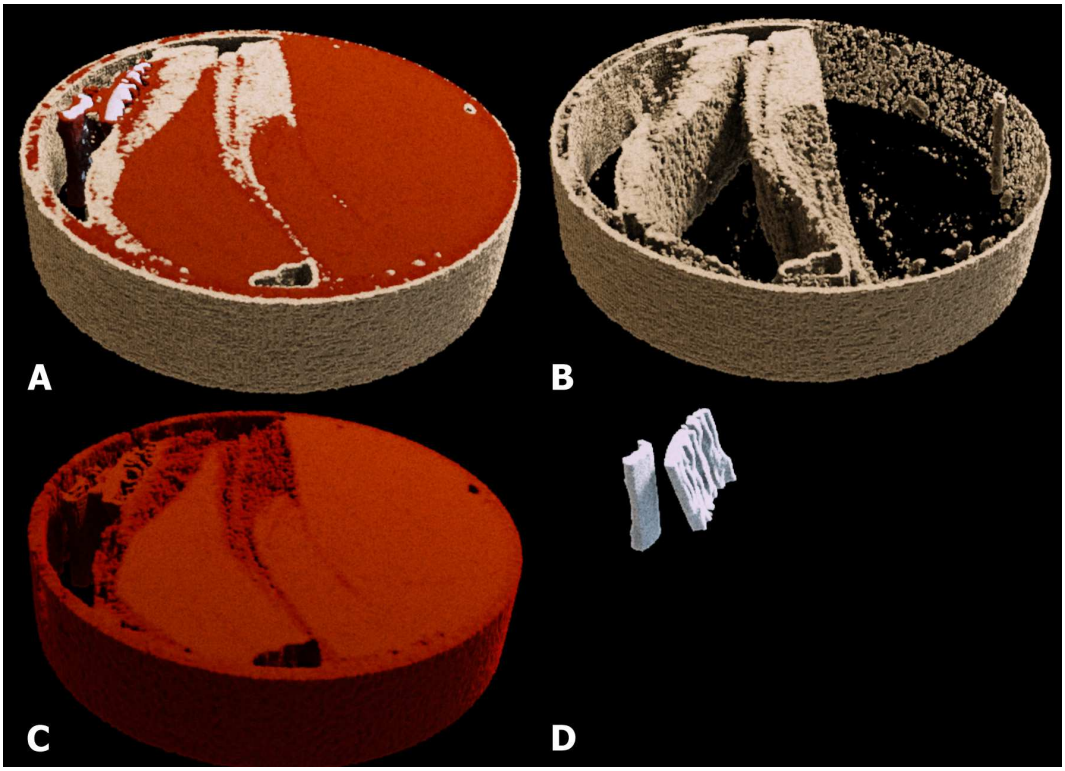


Figure 4.24: Occlusion during visualisation of a spectral CT dataset (Meat1127, section 9.3). **A**: the soft tissue (red), lipid (off-white/beige) and calcium (white) channels rendered simultaneously. Calcium is mostly occluded by the other materials. **B**: lipid only. **C**: soft tissue only. **D**: calcium only.

In this example, only a small portion of the calcium volume is visible in Figure 4.24A. Occluding structures may be removed by hiding the soft tissue and lipid channels (Figure 4.24B and C), and by only displaying the calcium channel (Figure 4.24D). However, this is a crude approach that also removes the context, which was previously provided by the other materials surrounding the ROI.

Therefore, hiding the occluding materials (or adjusting their transparency) during visualisation is a feasible solution, but the methods of minimising the undesirable effects of occlusion should also be studied. Ideally, the user should be able to select an ROI to be displayed without occlusion, while preserving the context. Chapter 7 demonstrates how the properties of spectral CT datasets (in particular, the classified material volumes created by MD) enable the creation of three tools for reducing occlusion.

4.4.5 Volume data processing and noise suppression

Section 3.3 has described the image quality issues that currently affect MARS datasets. In particular, streak, ring and beam hardening artefacts affect energy volumes, and some artefacts are also translated into material volumes. This points to the next requirement, namely the need for tools for suppressing noise and editing volume data during visualisation.

Such tools are necessary for current MARS users because the existing artefacts interfere with visualisation. This differs from the requirements of clinical users of conventional CT systems, who do not perform such processing. Instead, artefact removal or reduction is performed automatically by the data processing toolchain.

The methods and tools for real-time volume data editing and artefact removal are described in section 8.5.

4.4.6 Integration of visualisation and measurement tools

Visualisation of medical datasets is an interactive process that may also involve analysing the size and composition of ROIs. Measurement functionality is considered to be an essential feature of DICOM viewers [79, 81, 231, 238]. In addition, early studies into the applications of spectral CT indicate that users are interested in measuring the size and composition of features of both energy, and material volumes [9, 55].

However, the two existing custom applications for visualising MARS data (MARSCTExplorer and MARSCUDAVR) did not possess measurement capability [74, 129]. Therefore, the combination of visualisation and measurement, as shown in Figure 4.25, was not possible, and measurement had to be performed using external software such as ImageJ. This meant that the users who had found interesting features using the DVR visualisation implemented in MARSCTEx-

plorer and MARSCUDAVR had to find them again using a different program that was only capable of 2D slice visualisation.

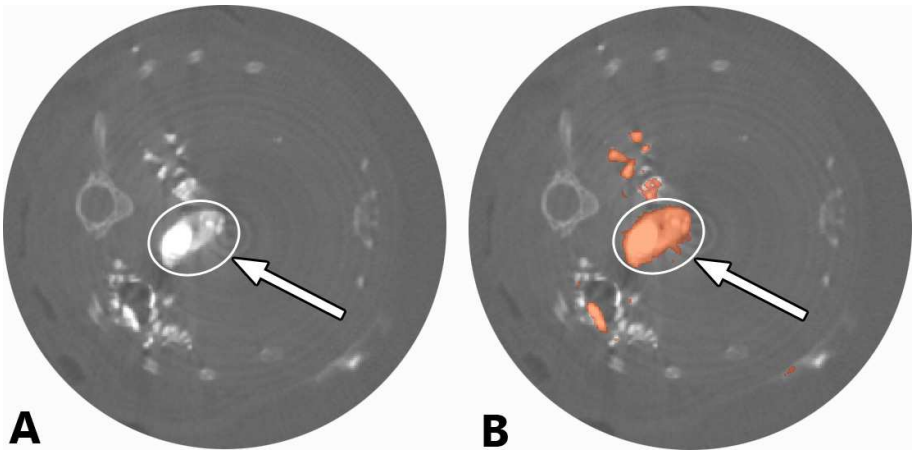


Figure 4.25: Two kinds of measurement that can be performed on spectral CT data. **A**: measuring the attenuation inside an ROI (marked with an arrow). **B**: measuring both material, *and* energy information. The third case, where material information alone is being measured, is not shown in this figure. In all three cases, measurement may include calculating basic statistics, such as the mean, median, minimum and maximum values, as well as the standard deviation of the pixels inside the ROI.

This should be considered a serious limitation of the older MARS visualisation software, as visualisation and measurement are two tasks that complement each other during the study of a spectral CT dataset. Visualisation allows the user to perceive the structure and location of ROIs, while measurement provides detailed information about their size and composition.

MARS spectral CT datasets contain both energy, and material volumes, which provide different information (anatomical and molecular, respectively). Therefore, just as the visualisation algorithms for spectral CT data must be able to render energy and material volumes, the measurement tools must also support both data types.

In conclusion, the integration of these two tasks is an obvious requirement, as it accelerates the user's workflow by reducing the number of tools that need to be used. Section 8.1 of this thesis describes the measurement tools created over the course of this research and the integration of these tools into a spectral CT data visualisation application.

4.4.7 Graphical user interface

Spectral CT is undergoing a transition from pre-clinical to clinical imaging. Therefore, a custom GUI for spectral CT data visualisation tools should incorporate the features desired by both scientists, and clinicians. However, the GUI design should attempt to address the needs of scientists first and foremost, as they currently comprise the user group conducting the most research using spectral CT.

We can refer to the classification by Escott and Rubinstein [238], who have identified the distinguishing features of GUIs of scientific and medical visualisation applications. Medical ("PACS-like", according to Escott and Rubinstein's classification) image viewers have very similar and consistent user interfaces. An example is shown in Figure 4.26. There are several reasons for this.

Figure 4.26: A PACS-like GUI (RadiAnt DICOM viewer [239]), showing multiple 2D views in a grid, along with measurements and patient information.

First, there is a limited and well-established set of tasks commonly carried out by most clinical users. Tasks usually include navigating through slices of a dataset, visually identifying anatomical structures, adjusting window and level settings, and performing measurements of ROI size or composition.

Second, the capabilities of medical imaging modalities, such as CT, PET, or MRI, are well-known. Special tools can be designed to take advantage of the

information provided by each modality. Presets for 2D (window and level settings, colour look-up tables) and 3D (transfer functions) visualisation can also be created.

Third, data is normally stored in the DICOM file standard and transmitted using the DICOM networking protocol. Therefore, integration with a PACS server (and, perhaps, with other, optional systems, such as the Radiology Information System, or RIS) is an important feature of the software.

On the other hand, scientific image processing and visualisation does not follow a standardised workflow. Therefore, the flexibility of the visualisation software and the number of features it possesses are the primary requirements. This means that the GUIs of scientific visualisation tools are more multifunctional and complex, as shown in Figure 4.27. These GUIs are closer to image editing applications such as Adobe Photoshop than to PACS-like image viewers. The datasets often need to be processed manually, and an advanced, flexible, image editor is more valuable than a simple but standardised tool with limited functionality.

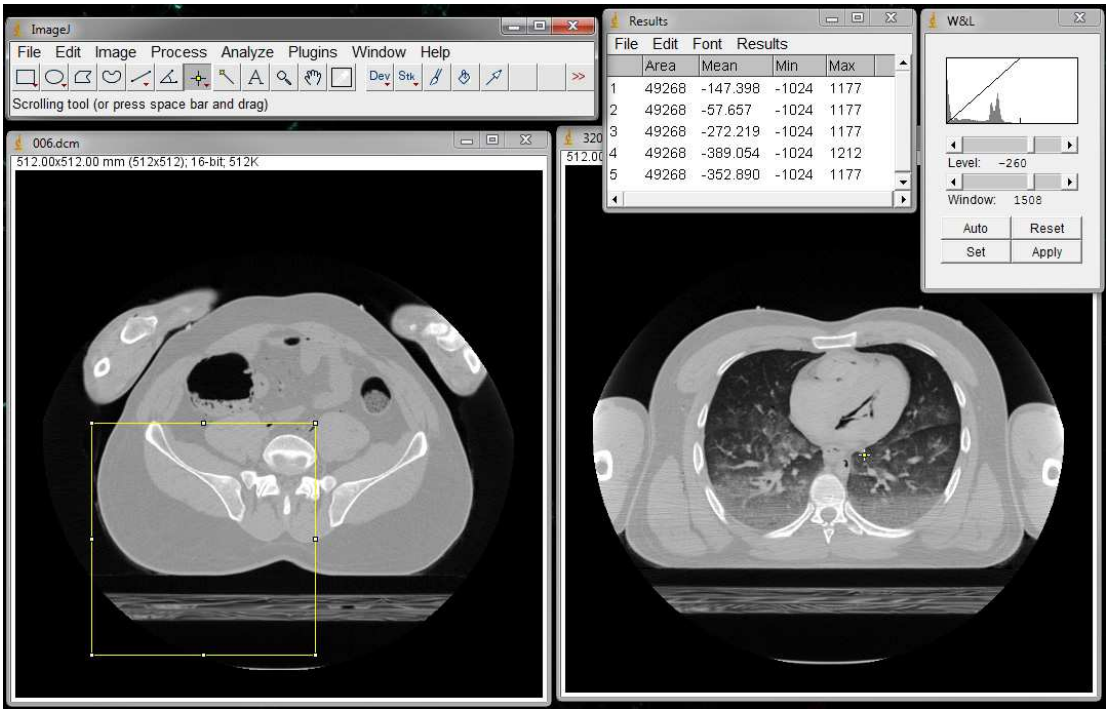


Figure 4.27: The GUI of ImageJ, a scientific image processing tool. Note the multiple windows, with numerous tools for performing image processing (available through the menu shown on the top left of the image).

At the current stage of development, there is no standard format for spectral

CT data and this technology has not yet been introduced into clinical practice. Therefore, a PACS-like GUI is insufficient. Considering the needs of current MARS users (who are mostly researchers into the clinical applications of spectral CT), we must focus on the design of a GUI that combines the features of both medical, and scientific image viewers.

4.4.8 Summary

Spectral CT improves on conventional and dual-energy CT by providing more detailed information about the molecular composition of the objects being scanned. However, most tasks conducted during the process of spectral CT data visualisation are similar across all three variants of CT. 2D slice visualisation, along with measurements of size or composition of ROIs, is essential, while 3D rendering can be used to visualise individual energy or material volumes.

The differences include the requirement for simultaneous visualisation of multiple volumes. This leads to the need for implementing custom algorithms for 3D visualisation and 2D slice data fusion. Volumes must appear visually distinct, which leads to the need for custom tools and GUIs for adjusting the visibility and appearance of multiple volumes. Traditional measurement tools need to be extended to provide additional material information to users of spectral CT data visualisation software. Finally, the tools for editing volume data and suppressing noise are also necessary at this stage of development of MARS technology.

These requirements are summarised in Table 4.1. However, there exist additional requirements that are not related to visualisation algorithms, the tools for working with spectral CT data, or the GUIs presented to the user. Nevertheless, they must be fulfilled by any application created for visualising MARS datasets, as they concern the integration of this application into the rest of the MARS toolchain. These requirements are summarised in Table 4.2.

Table 4.1: Summary of requirements for visualising current MARS datasets.

№	Feature	Reason for inclusion
1	Display of slices of a single energy or material volume using window and level settings	Standard medical and scientific technique for visualising slices of volumetric datasets.
2	Fusion of data from corresponding slices of multiple energy and/or material volumes	Required to show multiple materials concurrently, or to fuse energy and material data (by overlaying material information on top of a slice of an energy volume).
3	3D visualisation (for example, mesh rendering or DVR) that supports an arbitrary number of energy and/or material volumes	This technique is reasonably popular for showing the overall structure of the scanned object in scientific literature, and has also been used sparingly in clinical practice. Further, it is already familiar to MARS users.
4	A GUI for setting the colour, opacity, and visibility of each volume of a spectral CT dataset. This GUI should be targeted at users with no prior knowledge of transfer function design.	Required for making channels visually distinct during DVR and 2D slice data fusion, and for minimising occlusion during DVR. The feedback gathered from the MARS users pointed to the need for developing a simpler GUI that does not expose the full complexity of transfer function design.
5	Tools for visualising the location and structure of ROIs together with the context	MARS datasets often contain small ROIs that are occluded by other structures. Often, these ROIs are too small to be visualised on their own, and need some form of context to be present.
6	Volume data editing tools	MARS datasets often contain large amounts of noise and image artefacts that interfere with visualisation.
7	Generic measurement tools (for example, ROI size, number of pixels, statistics), and special measurement tools tailored to spectral CT data	Spectral CT provides both energy, and material information that can be measured. Measurement tools must be able to work with both data types, and must be able to measure any combination of energy and material volumes.

Table 4.2: Requirements for integrating visualisation software into the MARS toolchain.

№	Feature	Reason for inclusion
1	Support for reading DICOM files	The output of MARS reconstruction and material decomposition algorithms (energy volumes and material volumes, respectively) is stored using the DICOM file format.
2	Support for communicating with PACS servers using the DICOM networking protocol	All MARS datasets are stored on a PACS server, and using the DICOM networking protocol is the only way of communicating with it.
3	Support for datasets in the 16 bit-per-pixel format	The MARS team currently stores its energy and material volumes in this format to maintain the a large dynamic range.

4.5 Summary

This chapter reviewed the methods, algorithms and tools previously used for visualising spectral CT datasets. Aside from the work by members of the MARS project, no research dedicated to the theory or the requirements for spectral CT data visualisation could be found.

Therefore, I have formulated a set of requirements by analysing the basic properties of spectral CT technology, the properties of current MARS datasets, and the known clinical applications of spectral CT. The requirements include:

- A combination of 2D and 3D visualisation, with custom data fusion algorithms. These algorithms are described in Chapter 5.

- Editors for designing transfer functions for both energy, and material volumes. Chapter 6 describes the concept and implementation of a novel transfer function editor for working with both data types.

- Methods of reducing the effects of occlusion during 3D visualisation of spectral CT datasets. Chapter 7 discusses the concept and implementation of three such techniques.

- Various measurement tools that present relevant energy and/or material information to the user for analysis. Section 8.1 describes the extension of

traditional CT measurement tools for use with spectral CT data.

- Techniques for eliminating or suppressing noise and artefacts during visualisation, and methods for editing volume data in general. These are discussed in section 8.5.

In conclusion, considering the current state of MARS data, and spectral CT technology in general, a large range of functionality should be provided to the user. As the clinical applications of spectral CT become firmly established, specific requirements can be formulated for each individual application. This will likely happen at a later stage of development, as discussed in Chapter 10.

Chapter V

MARS Vision

This chapter describes the design and implementation of MARS Vision: a spectral CT data visualisation application based on the requirements established in Chapter 4. The main focus of this chapter is on the design and optimisation of a multi-volume DVR algorithm and a novel 2D slice data fusion algorithm. Particular attention is paid to preserving colour gradients and minimising colour distortion during blending.

In addition to visualisation algorithms, this chapter explains the architecture of MARS Vision and its integration into the MARS data processing toolchain. Other tools implemented in MARS Vision are described in subsequent chapters: the transfer function editor in Chapter 6, the tools for reducing occlusion in Chapter 7, and the tools for performing measurements, and editing volume data in Chapter 8.

This chapter is structured as follows. First, a high-level overview of the architecture and GUI of MARS Vision is given in section 5.1. The supported file formats and loading procedures are explained in section 5.2. Section 5.3 discusses the implementation of a novel DVR algorithm for spectral CT data. Section 5.4 describes the implementation of the data structure that enables interaction with 3D visualisation (voxel selection, also called picking). Section 5.5 details the optimisations required to achieve interactive DVR performance. Lastly, section 5.6 focuses on 2D slice visualisation. It explains the need for 2D data fusion and presents an algorithm that dynamically fuses slices of multiple volumes, using transfer functions to assign colour and opacity.

5.1 Overview

MARS Vision is an application written specifically for the purpose of illustrating the visualisation algorithms, tools, and GUIs developed over the course of this research. It is the last stage in the MARS software toolchain, and contains the

tools necessary for working with currently-available MARS spectral CT datasets. However, it is also an extensible framework that provides a platform for future research into the visualisation of spectral CT data.

The core of MARS Vision is based on an application called Exposure Render, which has been created by Kroes et al. to demonstrate the practical implementation of a novel Monte-Carlo ray tracing (MCRT) DVR algorithm [103]. I have decided to convert Exposure Render into a spectral CT visualisation application because:

- It is published under the BSD software license, which allows unrestricted modification, as well as commercial use. The changes to the source code do not need to be made public. This is an important requirement, as the MARS project is involved in the commercialisation of spectral CT technology.

- It implements a DVR algorithm that provides high visual quality at interactive frame rates. This algorithm is unusual compared to many other implementations of DVR because it simulates the transfer and scattering of light emitted by external light sources. This simulation produces shadows, which are known to be a useful depth cue that can help the user understand the relative positions of features in objects with a complex geometry [240, 241, 242, 243].

- It has a relatively small code base (around 120 classes, totalling around 20000 lines of source code). Applications such as MITK [69] and 3D Slicer [68] have also been considered as possible platforms for developing spectral CT data visualisation software. However, the code bases of these applications are large and difficult to modify. For example, 3D Slicer consists of around 100 separate components (each comprising multiple classes), such as the core rendering algorithms, GUI and dozens of tools and plug-ins, most of which are not applicable to spectral CT data visualisation. Therefore, I have decided that it is easier to start from a simple, single-purpose volume visualisation application, and adapt it for working with MARS spectral CT datasets.

- The core of Exposure Render is written in C++, the DVR algorithm is implemented in CUDA (described in section 2.3), and the GUI uses the Qt

toolkit [244]. All languages and libraries are cross-platform. This is important for the MARS project, which uses a mixture of Microsoft Windows, Apple OS X and GNU/Linux workstations.

However, there are several notable disadvantages:

- Exposure Render only supports single-volume rendering, which means that its GUI contains no elements for performing data fusion. Therefore, a large amount of work needed to be performed to convert it into a native spectral CT data visualisation application. This was done by re-designing its GUI and implementing 2D and 3D data fusion algorithms, as described in this chapter.

- It implements an unusual stochastic rendering algorithm that generates a high amount of noise. Therefore, a noisy image is shown to the user whenever rendering restarts (this happens when a parameter such as the transfer function or the camera position is updated). The noise is quickly suppressed as more and more iterations are rendered, but, for the first few iterations of the algorithm, the image quality is poor. This issue is further discussed in section 5.3.2.

- It is unknown how Exposure Render's MCRT DVR algorithm will scale as the volume size increases and bricking or level-of-detail rendering techniques become needed. Multiple bricking or level-of-detail rendering techniques have been proposed for standard ray marching DVR algorithms [245, 246, 247]. However, no work has been conducted into the incorporation of these techniques into an MCRT DVR algorithm.

5.1.1 Architecture

The architecture of MARS Vision is designed to provide a modular platform for future tool development. Therefore, the monolithic design of Exposure Render had to be split into a number of subsystems for controlling the rendering process, managing volume data, updating transfer functions, and so on. The three main subsystems that can be identified are: the core, the GPU rendering algorithms, and the suite of tools, as shown in Figure 5.1. This section briefly describes

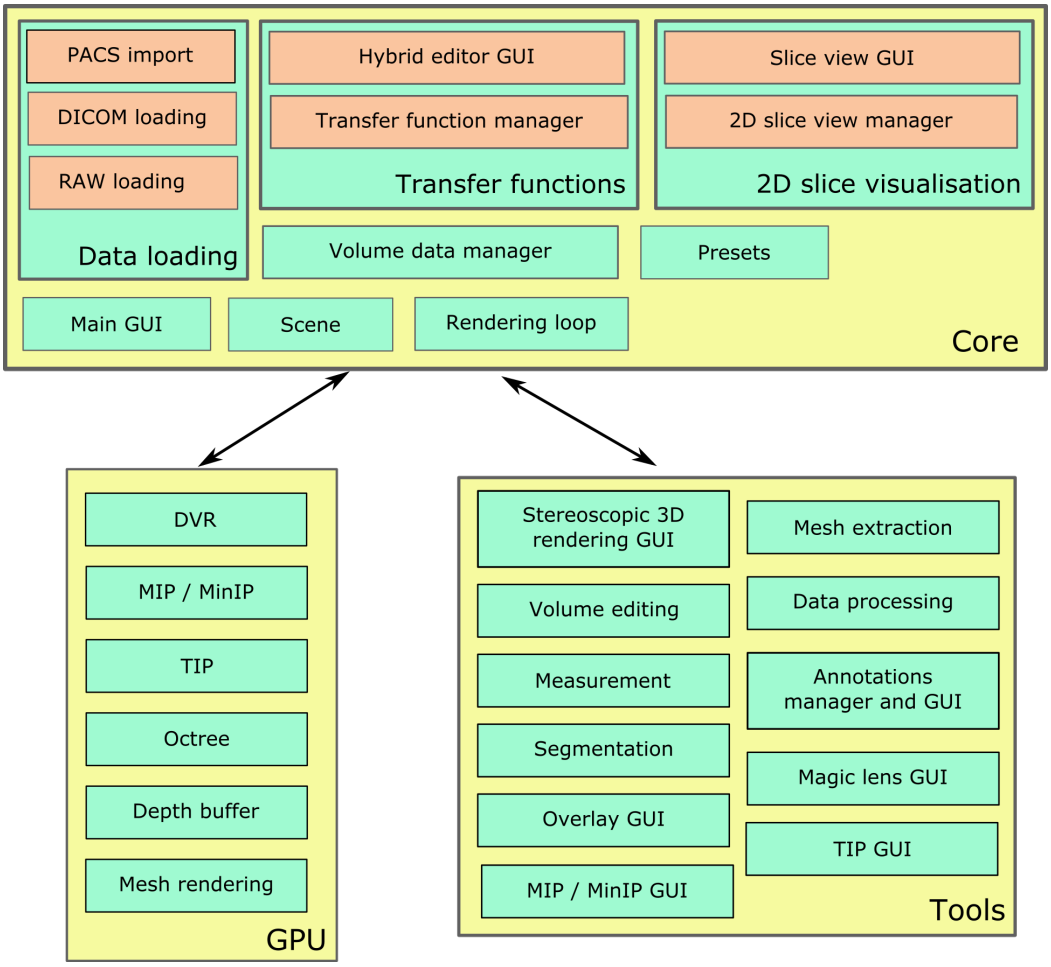


Figure 5.1: A high-level block diagram of MARS Vision, showing the three main subsystems: the core, the GPU rendering system, and the optional tools suite.

the functionality of each subsystem and provides references to sections describing individual elements in greater detail.

The core subsystem is responsible for tasks such as saving and loading data, adding new volumes or removing existing ones (through the volume manager) accepting the user's input, maintaining the state of the transfer functions (through the transfer function manager), and saving presets. The core controls the 3D rendering loop through the *scene* object, which contains all parameters necessary to render a dataset. All rendering calls are issued by the rendering loop and sent to the GPU subsystem, which then executes the appropriate algorithm.

Supported 3D visualisation techniques include DVR (section 5.3), maximum, minimum, and threshold intensity projections (MIP, MinIP and TIP, as described

in Chapter 7), and mesh rendering (section 10.2.5). The GPU subsystem is also responsible for generating and maintaining the octree used for empty-space skipping (section 5.5) and the depth buffer, which is required for interacting with DVR visualisation (section 5.4).

The core and the GPU subsystems provide all functionality necessary to visualise a spectral CT dataset using 2D or 3D techniques. All tools for analysing data, removing image artefacts, segmenting volumes, or placing annotations, are grouped in a separate subsystem. This subsystem is optional, because it is built on top of the core MARS Vision framework, and is not strictly required for visualisation (however, these tools can help users interact with spectral CT data, as described in Chapters 7 and 8). The tools are intentionally decoupled from the core rendering algorithms, data managers, and data structures. This is done to improve the extensibility, modularity, and maintainability of MARS Vision.

One important property of MARS Vision's design is that it does not allow the users to create a custom chain of data processing operations and arbitrarily modify the source volume data. This is intentional, as the aim of the MARS project is to create a software product suitable for users with little technical skill or experience using visualisation or data processing tools. Currently, some advanced image processing functionality must be supported, as discussed in 4.4.7; however, the long-term goal is to create a specialised but limited user interface similar to DICOM viewers. Therefore, the range of possible actions is restricted by design. This is partially due to the earlier experiences with MARSCTExplorer, which gave the users the freedom to arbitrarily intermix volume data, but was only usable by visualisation experts, as discussed in section 4.3.2.1.

5.1.2 Main GUI

The GUI of MARS Vision has been designed to offer a compromise between a medical and an image editing GUI (as described in section 4.4.7). A balance had to be found because an image editing GUI is powerful, but generic (that is, designed to be used for working with many types of images and volumetric datasets), while a medical GUI is too specialised (customised to support a particular workflow and standardised data types), but well-suited for a certain subset of tasks.

Currently, spectral CT is in a transitional stage of development. Most users are scientific researchers, but the composition of the user base is likely to change in

the next few years, as spectral CT is introduced into clinical practice. The GUI of MARS Vision, shown in Figure 5.2) reflects this situation by addressing the needs of researchers, but also incorporating several concepts from medical GUIs.

For example, the tools for editing and processing volume data (further described in sections 8.4 and 8.5) and the GUI elements for changing the volume rendering settings are currently required for studying MARS spectral CT datasets. On the other hand, features such as a grid of 2D slice views, and annotations overlaid onto 2D slices are commonly found in DICOM viewers (for example, eFilm-Lite [63], InteleViewer [66], RadiAnt [239], DicomWorks [238], Osirix [248], or Vantage PACS [80]). Therefore, MARS Vision implements a hybrid of the two GUI styles.

Two GUI layouts are shown in Figure 5.2, though other layouts are also provided. By default, the widgets are arranged in a 2×2 grid (bottom image) to imitate the appearance of standard DICOM viewers, and to maximise the amount of space available to display the data.

Most panes, such as all slice view widgets (Figure 5.2E), tools (Figure 5.2C), and the transfer function editor (Figure 5.2F), can be detached to become free-floating windows. This behaviour is similar to that of ImageJ, the software commonly used for spectral CT data analysis by members of the MARS team.

All 2D GUI elements are operated by manipulating standard widgets such as text boxes, buttons, slider and spinner controls and drop-down lists [249]. The operation of such controls is trivial and does not need to be explained in detail.

The user is able to interact with the 3D visualisation (Figure 5.2D) by directly clicking on visible surfaces. This interaction is enabled by a data structure called the depth buffer, which maps 2D mouse click coordinates to 3D voxel coordinates. The depth buffer is explained in detail in section 5.4.

5.1.3 Summary

In conclusion, MARS Vision is a modular and extensible visualisation application, which is an important requirement for the MARS project. This software is currently being used to visualise spectral CT datasets and develop novel visualisation and interaction techniques. It may serve as a base for the development of visualisation software for use in clinical settings.

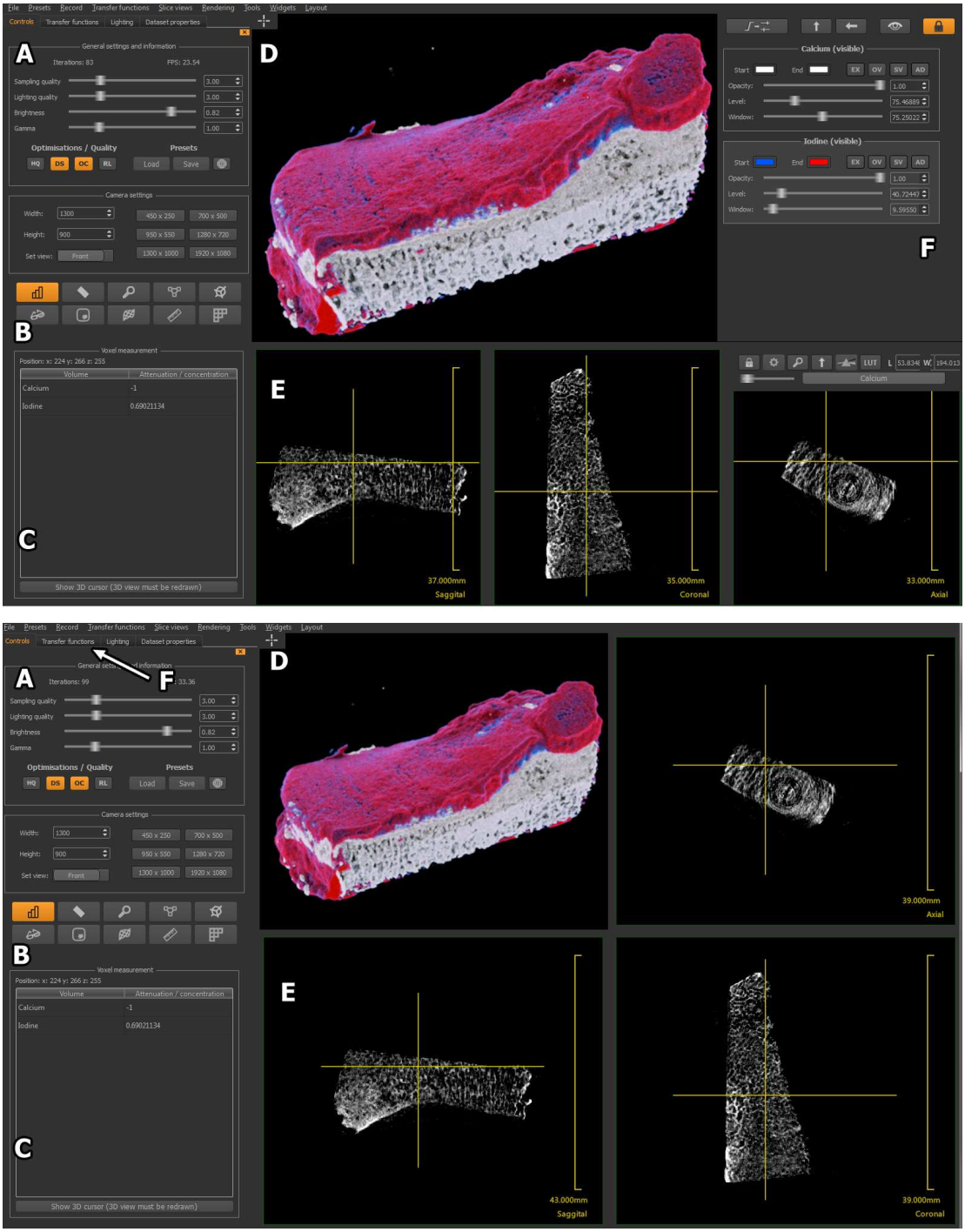


Figure 5.2: Two MARS Vision GUI layouts. **A**: image quality settings such as sampling step size and brightness. **B**: tool selection buttons. **C**: currently-selected tool. **D**: 3D volume visualisation. **E**: 2D slice views. **F**: A list of two hybrid transfer function editors (described in detail in Chapter 6) in the simple form. The list can be activated by moving the mouse cursor over the right side of the 3D canvas (top image). Alternatively, the list can be moved into a tab on the left side of the GUI (bottom image).

5.2 Data loading and internal storage

MARS Vision can load datasets stored in two different formats: DICOM [250], and a binary volume data format. The former is a standard format used throughout the current MARS data processing toolchain, while the latter is required for legacy support. It is explained in detail in section 5.2.3.

Section 3.1 has explained that the MARS project is currently re-designing its data processing toolchain to store all energy and material volumes in the DICOM file format. However, the option to import raw binary data is necessary for supporting older MARS datasets that were acquired and processed before the introduction of the DICOM-based toolchain. This option is also useful for testing MARS Vision using the volumetric datasets that were not processed by the MARS toolchain.

5.2.1 Internal storage

Internally, MARS Vision allocates an array in RAM for each volume and stores volume data in the 16-bit unsigned integer format. Therefore, the dynamic range of each voxel is $2^{16} = 65536$. Ideally, a format such as the IEEE754 64-bit or 32-bit floating-point [251] should be used to store data, as it provides higher precision and larger dynamic range. However, MARS Vision, and the rest of the MARS toolchain do not use this format for two reasons:

- The precision and range offered by storing voxel values in a floating-point format is rarely required for medical visualisation, where the hardware used for image acquisition is usually the limiting factor. For example, the vast majority of commercial single-energy CT systems store data in the Hounsfield Unit format (section 3.2.1). The Hounsfield scale has no upper limit, but the typical HU range that corresponds to tissues inside the human body is around 4000, with -1000 HU corresponding to air, and around 3000 HU corresponding to dense bone.

  Therefore, a 16-bit integer format is sufficient to preserve the dynamic range for conventional CT imaging. Following this convention, the energy and material volumes produced by MARS reconstruction and material decomposition software also store data in this format. Currently, it is unknown

whether this approach will need to be revised as spectral CT imaging develops.

- MARS Vision's DVR algorithm is executed on the GPU. However, the amount of GPU VRAM is limited (usually restricted to around 4 GB, as discussed in section 2.2.1). Therefore, it is impractical to use GPUs to visualise volumetric datasets stored in the 32-bit floating-point format. Instead, data formats that use 8 or 16 bits per pixel are preferred.

5.2.2 Loading of binary files

MARS Vision is able to load volumes converted to a raw binary format, accompanied by a separate descriptor file. This file contains several essential attributes, such as the voxel size (size of each voxel, in millimetres), dataset dimensions (number of voxels), data format (signed or unsigned integer, 8, 16, or 32 bits per voxel), and the path to the binary file. The descriptor is necessary, as the raw volume data file contains no metadata.

Most older MARS datasets are stored as stacks of TIFF files. Image processing software such as ImageJ can be used to open a stack of files corresponding to each volume, and save it as a raw binary file. In this case, the user must manually create a descriptor file.

To address this problem, a conversion script is included with MARS Vision. The script automatically reads all TIFF files in each stack, and saves both the raw binary files, and the corresponding descriptor files.

The loading of binary files is gradually being phased out in favour of loading DICOM image series. New MARS datasets are no longer saved in the TIFF format, which eliminates the need for conversion. However, raw binary data import capability is still an important feature, as it allows for the loading of volumetric datasets that have not been created by the MARS software toolchain. This includes datasets from public repositories, such as the Volume Library [77].

5.2.3 Loading of DICOM series

The MARS software toolchain stores all of the reconstructed energy volumes for a dataset as one DICOM series. All additional information is stored in tags, which are present in every DICOM file output by MARS reconstruction and material

decomposition software. This metadata includes the energy bin ranges, the acquisition parameters, the name and date of the scan, and so on (refer to the PhD thesis by de Ruiter [59] for a detailed description of the parameters stored in the DICOM files produced by the MARS software toolchain). Therefore, separate descriptor files are no longer necessary.

A DICOM series consists of a number of DICOM files, where each file represents one axial slice through the dataset. Data from all energy volumes acquired during the scan is stored in *frames* inside each file. Frames, as shown in Figure 5.3, provide a way of compactly storing multiple images in the same file. In this example, the data from three energy volumes is packed into a single pixel array.

The DICOM file format uses the *scale* and *intercept* parameters to control the dynamic range of the data. However, different scale and intercept parameters for each frame are not supported. Therefore, the MARS reconstruction software must scale all energy volumes to the same dynamic range. In practice, this does not pose a problem, as the minimum and maximum attenuation values are very similar across all energy volumes [59].

However, the same cannot be done for material volumes, as the concentrations

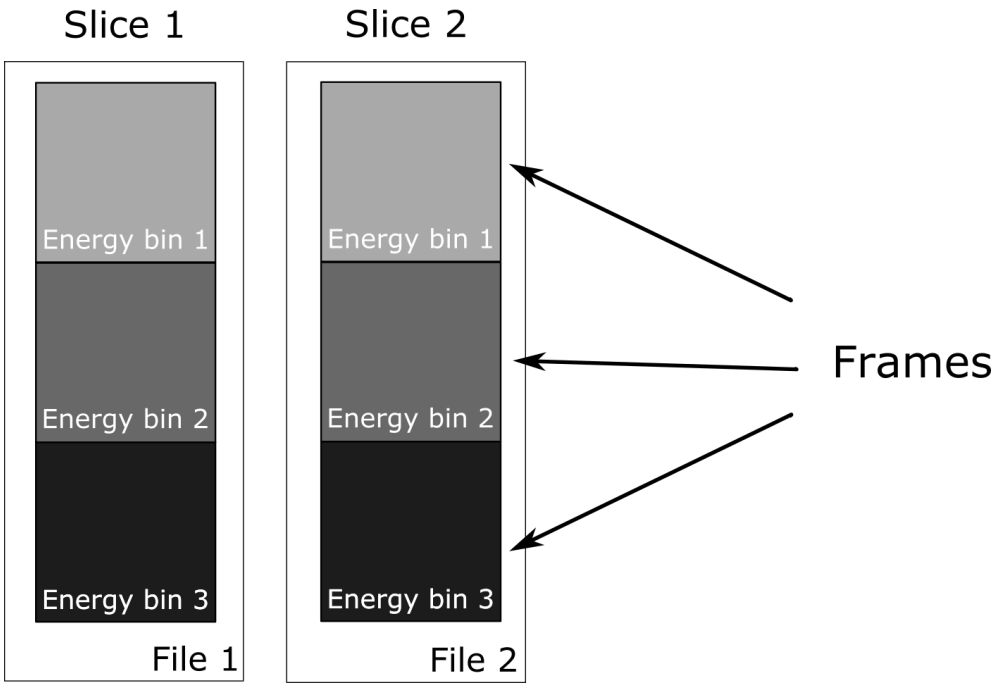


Figure 5.3: Storing multiple energy volumes using DICOM frames. Each DICOM file packs data from all energy volumes into a single pixel array.

of different materials can vary substantially. For example, the concentration of one material may range from 0 to 10 mg/ml, while another material may range from 0 to 100 mg/ml. Scaling all materials to the largest range (which must be done if the the same DICOM series is used) will lead to a loss of information, as the dynamic range for some materials will be compressed. Therefore, every material is scaled separately and stored in a different DICOM series.

MARS Vision includes a custom GUI for loading multiple DICOM series, as shown in Figure 5.4. This GUI can be used to import a combination of energy and material volumes, as long as the dimensions of each series are identical. DCMTK [252] is used to load DICOM files and parse the tags.

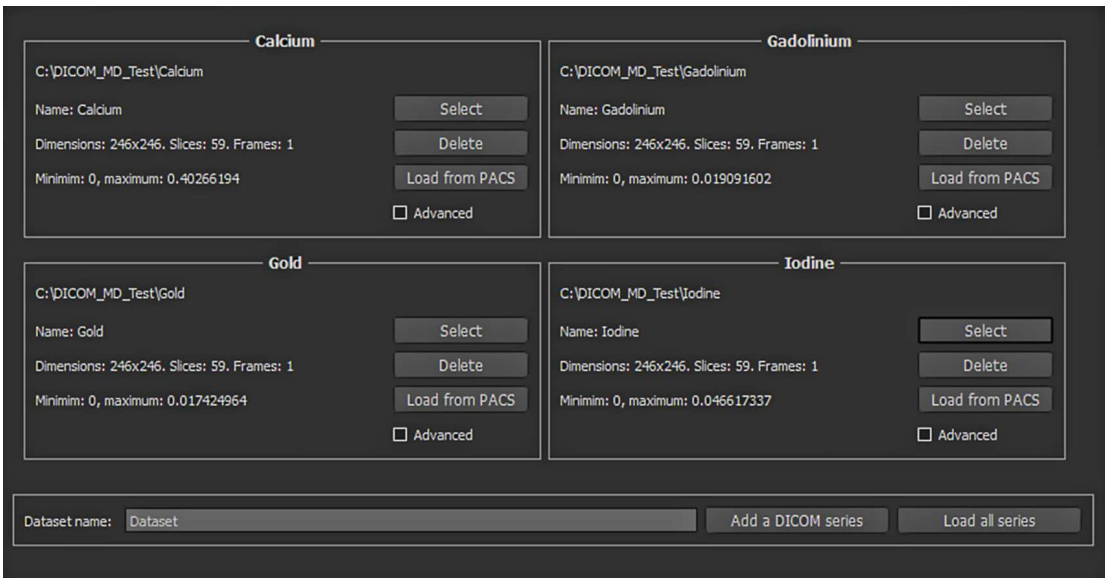


Figure 5.4: The GUI widget for importing multiple DICOM series. This example shows four materials stored as separate DICOM series inside different folders. Note that each material has a different dynamic range.

5.2.3.1 Loading of DICOM series from PACS

The integration of visualisation software with the rest of the toolchain is an important requirement for the MARS project. The software must be able to access the PACS server and retrieve any correctly-formatted DICOM series.

MARS Vision accomplishes this by providing a PACS import GUI, shown in Figure 5.5. The user is able to perform standard queries for a particular patient name, study or series description, scan acquisition date, and so on. This GUI and

the underlying DICOM network communication code has been written by de Ruiter as part of his research into improving the MARS data processing toolchain [59], while I have integrated it into MARS Vision.

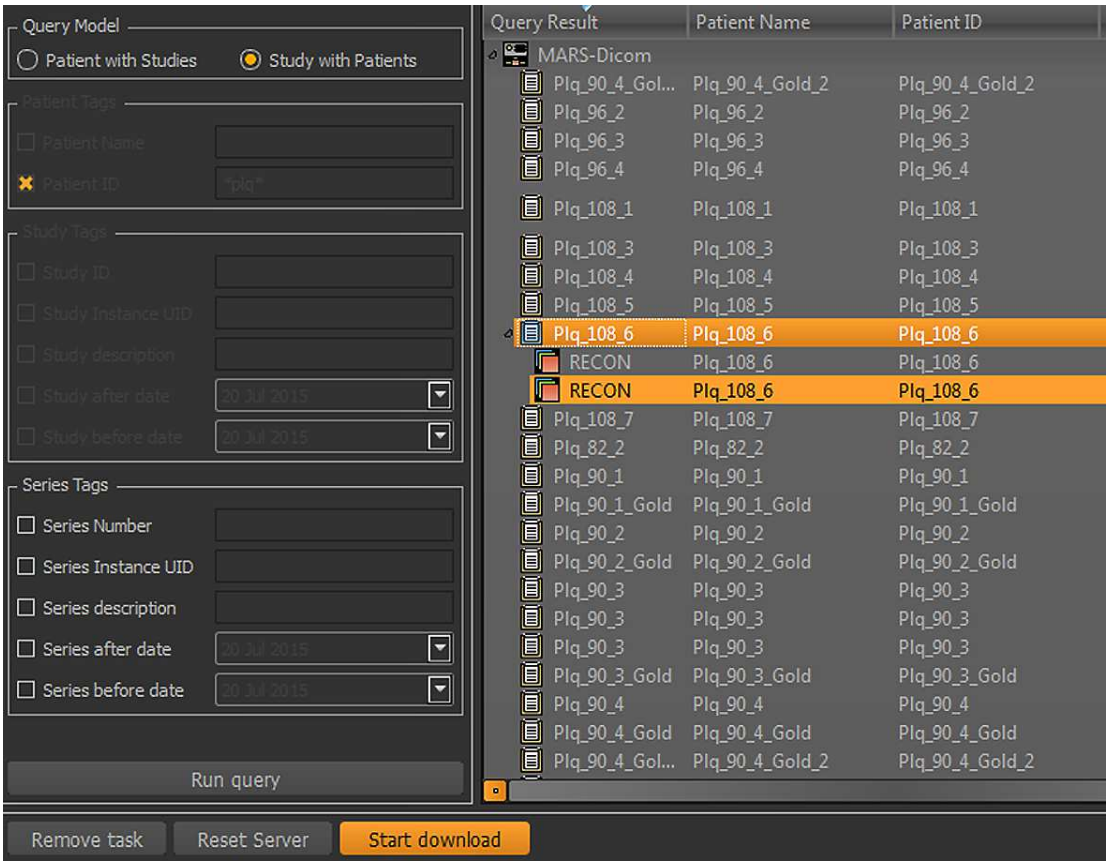


Figure 5.5: The GUI for importing a DICOM series from a PACS server.

5.2.4 *Summary*

MARS Vision is compatible with the standard DICOM file format. It is able to load DICOM datasets from the local hard drive, or from a networked PACS server. DICOM is the preferred format, as it allows MARS Vision to parse the tags contained inside the files, and provide all relevant information about the dataset to the user.

Another option is to load a dataset stored in the TIFF file format by converting stacks of TIFF images into raw binary files and special descriptor files. This option is included for legacy support, as older MARS datasets (stored in the TIFF format) are still being studied by the MARS group.

5.3 Volume rendering

This section discusses the original single-volume DVR algorithm implemented in Exposure Render, as well as the modifications implemented in MARS Vision to support multi-volume rendering. The modifications that enable interaction with volume data are explained in section 5.4, while the optimisations applied to this DVR algorithm are described in 5.5.

5.3.1 The original Exposure Render DVR algorithm

Exposure Render implements an unusual DVR algorithm, which is referred to as a Monte-Carlo ray tracing (MCRT) DVR algorithm by Kroes et al [103]. It uses a stochastic Monte-Carlo approach to sampling, as opposed to the traditional technique of ray marching (described in section 2.2.2). This algorithm, along with the modified version implemented in MARS Vision, is written in CUDA (section 2.3.3).

This algorithm progressively refines the image by performing a sequence of steps called *iterations*. Each iteration consists of the following steps:

- Stochastic raycasting. Each sampling ray finds a scattering point inside the volume and calculates the illumination at that point using a single randomly-chosen light source. This step generates a low-quality, noisy *frame estimate* image.

- Blurring of the frame estimate using a Gaussian kernel.

- Monte-Carlo integration of the frame estimate. The current frame estimate is added to the sum of all frame estimates, called the *running estimate*.

- Tone mapping. This step performs colour correction and adjusts the brightness based on the settings selected by the user.

- Denoising (using the K Nearest Neighbour filter [253]) and drawing of the final running estimate buffer to the screen.

Every iteration refines the image displayed to the user. Gradually, a high-quality image is built up, as shown in Figure 5.6. Kroes et al. have demonstrated

that this algorithm achieves interactive performance (30-70 iterations per second) while rendering conventional CT datasets [103].

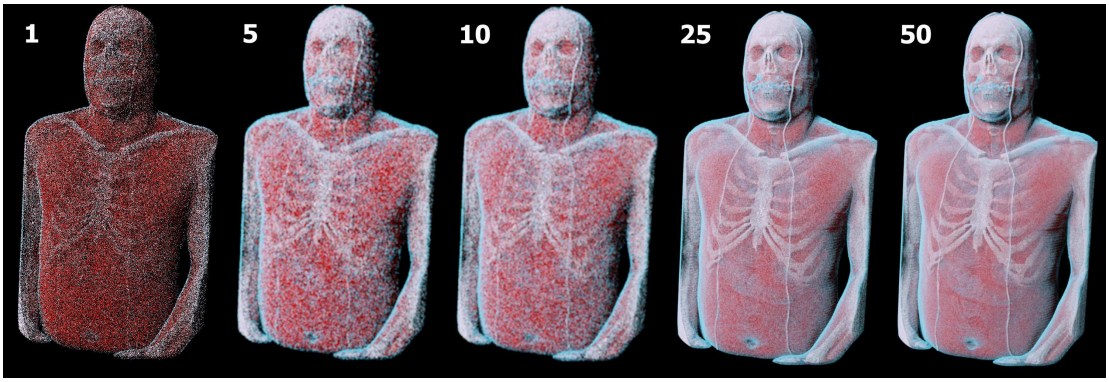


Figure 5.6: Progressive refinement of the image in Exposure Render and MARS Vision. The number of iterations is given next to each image.

The reason for using Monte-Carlo sampling is that it allows this algorithm to only calculate illumination once per iteration, as shown in Figure 5.7. Standard ray marching algorithms calculate it at every step along the ray; in contrast, Exposure Render first calculates the scattering position by sampling along the ray until a randomly-chosen density is reached, and then calculates the illumination at the scattering point.

This is an important distinction, as Exposure Render implements a highly complex illumination model. Depending on the gradient at the scattering point (calculated using the central differences algorithm), it chooses between using a phase function, or a BRDF lighting model. This is referred to as "hybrid scattering". The phase function is used inside homogeneous regions (where the gradient is close to 0), while the BRDF is used at the edges. Therefore, a reduction in the number of illumination calculations allows this algorithm to achieve interactive performance [103].

5.3.2 Overview of MARS Vision's DVR algorithm

This section describes the sequence of actions performed by the stochastic raycasting step of MARS Vision's DVR algorithm. This is the only step that must be modified to support multi-volume rendering using MCRT. This sequence is shown in Figure 5.8 and Algorithm 1.

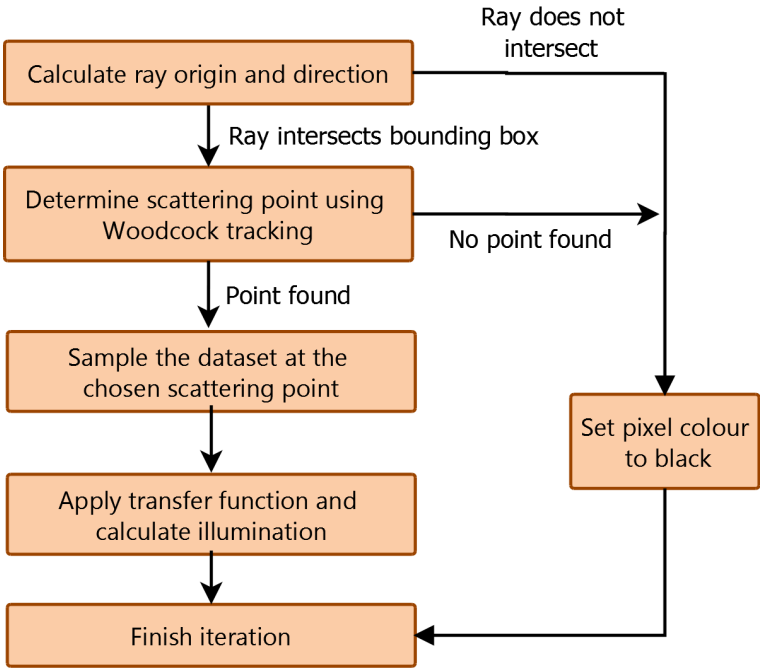


Figure 5.7: The sequence of actions performed by the stochastic raycasting step of Exposure Render's DVR algorithm.

To the best of my knowledge, this algorithm is the first to implement multi-volume rendering using MCRT. This is achieved by modifying the scattering point selecting step and the illumination step of the original single-volume MCRT DVR algorithm created by Kroes et al.

First, a scattering location is determined by sampling all visible volumes separately. From now on, the word "visible" will be implied, because a volume that has been hidden by the user is not involved in the rendering process. The point that is closest to the origin of the sampling ray is chosen and set as the scattering point for the current iteration. This step is described in detail in section 5.3.3.

Next, all volumes are sampled at the chosen point, and illumination is calculated for each volume. This step determines a colour value (in CIE XYZ 1931 colour space [254]), and an opacity percentage for each volume. Section 5.3.4 describes the illumination step in greater detail.

Finally, colour blending and opacity correction are performed. This generates the final pixel colour for the current iteration. These steps are described in section 5.3.5.

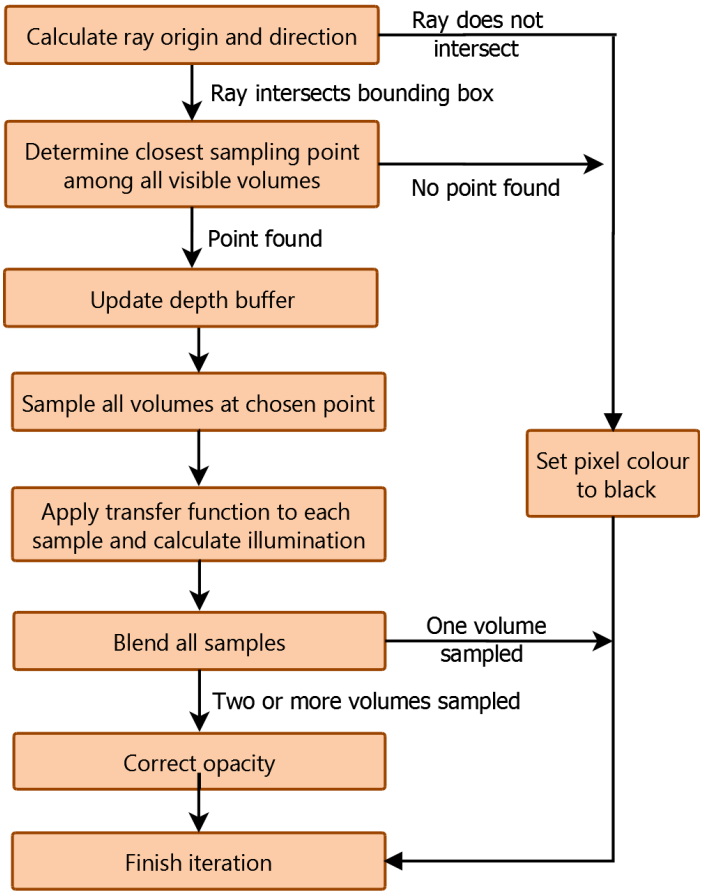


Figure 5.8: A single iteration of MARS Vision's multi-volume DVR algorithm.

This algorithm has no theoretical limit on the number of volumes that can be rendered. However, in practice, the amount of GPU VRAM available for storing volume data is the primary restriction.

5.3.3 Selection of the scattering position

The scattering point selection algorithm implemented in Exposure Render uses a method known as Woodcock tracking, or delta tracking [255]. The scattering point for a ray is selected as follows:

1. A random opacity value is calculated by generating a random number between 0 and 1 (0 and 100%) and multiplying it by the *maximum extinction*

Algorithm 1 A single iteration of MARS Vision's multi-volume MCRT DVR algorithm. Minor steps have been omitted for clarity.

Data: Ray R cast from pixel $P_{x,y}$ with origin R_o and direction R_d, n volumes $V_{1...n}$, n transfer functions $T_{1...n}$
Result: pixel colour C generated by ray R

initialization:
temporary colour TC
temporary opacity TO
main scattering position $S_{main} \leftarrow \infty$
temporary scattering position S_{temp}
total opacity $O \leftarrow 0$

determine the closest scattering point:
for $i \leftarrow 1$ **to** n **do**
  use Woodcock tracking to find scattering point S_{temp} in V_i ;
  if S_{temp} is closer to R_o than S_{main} **then**
    $S_{main} \leftarrow S_{temp}$
  end
end

update the depth buffer for P, using S_{main}

colour and opacity calculation:
if $S_{main} \neq \infty$ **then**
  for $i \leftarrow 1$ **to** n **do**
    $TO \leftarrow$ calculate opacity at position S_{main} in V_i using T_i
    $TC \leftarrow$ calculate lighting for position S_{main} in V_i using T_i
    $TC \leftarrow TC \cdot TO$ *multiply colour by opacity*
    $O \leftarrow O + TO$ *accumulate opacity*
    $C \leftarrow C + TC$ *accumulate colour*
  end
end

divide by the accumulated opacity to produce a weighted average:
if $O \neq 0$ **then**
  $C \leftarrow C \div O$
end

save C

coefficient set by the user. This coefficient is an arbitrary value ≥ 0; in MARS Vision, it ranges from 0 to 100. This is the target opacity that a sampling ray must reach by sampling the volume and accumulating opacity.

2. A ray-box intersection algorithm is used to find the point where the ray enters the dataset's bounding box, and the point where the ray exits it.

3. The ray starts sampling at the point of entry and iterates towards the exit point, sampling at every step. The size of the step is adjustable by the user.

4. At each step, the ray samples the volume and uses the transfer function to find the opacity at that point.

5. The opacity is divided by the maximum extinction coefficient and added to the sum over the entire ray.

6. The ray terminates when the target opacity has been reached. The position in the volume where this occurs is set as the scattering location for the current iteration. If the ray exits the bounding box, then the scattering location for the current iteration is declared as invalid.

As this sequence shows, the maximum extinction coefficient controls both the target opacity, and the contribution of each sample to the sum. If the coefficient is small (for example, close to 0), then the target opacity will be low (step 1), and the contribution of each sample will be large (step 5). Therefore, fewer samples will need to be taken to reach it. Such settings usually render a solid opaque surface, as shown in Figure 5.9A.

If the maximum extinction coefficient is large (for example, close to 100), then the target opacity will be high, but the contribution of each sample will be small. Therefore, a ray has a greater chance of going deeper into the volume, which creates the appearance of a translucent material. The larger the coefficient, the less likely the rays are to find valid scattering points. The volume, regardless of the transfer function, begins to resemble a cloud of gas (Figure 5.9B).

Woodcock tracking can be extended to sample multiple volumes that occupy the same region of space, with one restriction: the maximum extinction coefficients of all volumes must be the same. This means that all volumes must be treated as

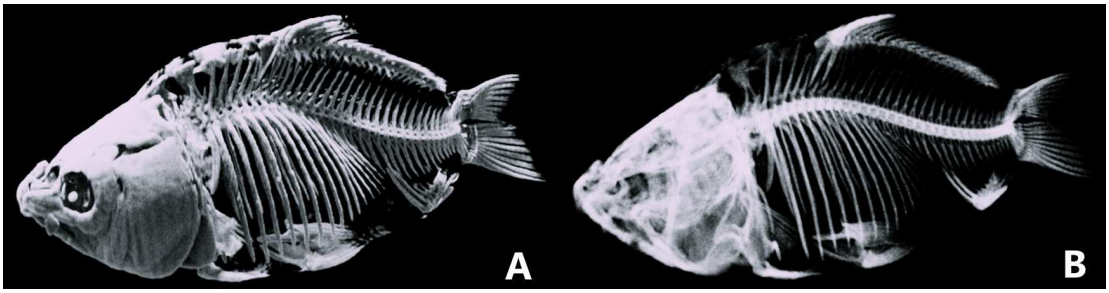


Figure 5.9: Rendering the Carp dataset using a low maximum extinction coefficient (**A**), and a high maximum extinction coefficient (**B**). The transfer function is the same in both cases.

materials of equal density. If the coefficients are different, then the target opacity (determined in step 1 of the algorithm above) cannot be calculated.

As described in section 4.4.4, precise control over the density of materials is one of the main methods of reducing occlusion during 3D rendering. For conventional CT data visualisation, this is done by assigning opacity to various data ranges using a transfer function. However, conventional CT datasets only comprise a single volume, which contains all materials. Therefore, there is no need for per-volume opacity settings, and conventional Woodcock tracking, as implemented in Exposure Render, is acceptable.

The spectral CT datasets produced by the MARS toolchain contain multiple energy and material volumes, and any combination of them may be visualised. For example, one volume may need to be displayed as a translucent outline, with another volume appearing as a solid surface. Therefore, it is essential that MARS Vision's multi-volume DVR algorithm supports a different opacity setting for each volume.

My extension of Exposure Render's single-volume Woodcock tracking algorithm solves this problem by sequentially applying it to find a scattering point in each volume, as shown in Figure 5.10B. The scattering point that is closest to the origin of the ray is set as the scattering location for the current iteration, as shown in Figure 5.11. This value is stored as an *(x,y,z)* coordinate in volume space.

This allows the maximum extinction coefficients to be set separately for each volume. In practice, this is done by adjusting the opacity slider contained in the hybrid transfer function editor GUI (section 6.4).

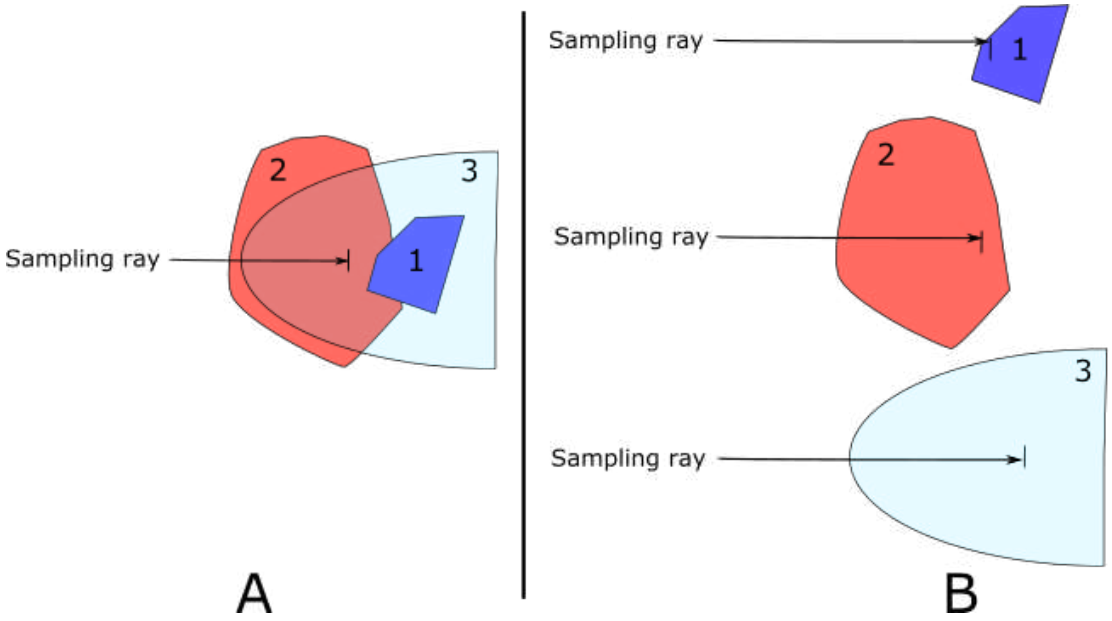


Figure 5.10: Differences between the traditional approach to sampling multiple datasets during DVR (**A**) and MARS Vision's approach (**B**). If multiple volumes (three, in this example) are being rendered, then the traditional approach samples all volumes simultaneously until a certain density is reached. MARS Vision samples all volumes separately, as shown in this figure, and then chooses the closest sampling point, as shown in Figure 5.11.

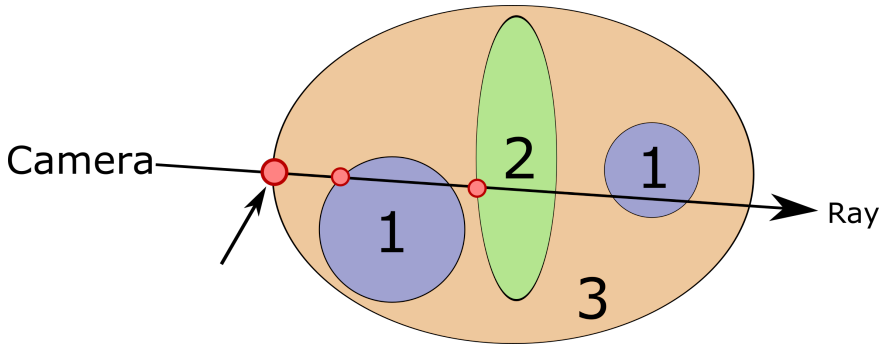


Figure 5.11: Selection of the scattering location in multiple overlapping volumes. In this example, three scattering points are found, and the closest one (marked by an arrow) is selected.

However, it must also be noted that this modification to the original Exposure Render DVR algorithm change the optical model used for volume rendering. The original model was single scattering, while the proposed model has not yet been described in literature. However, it still uses the same principle, namely the simu-

lation of the scattering of light emitted by an external light source [60]. Therefore, it may be considered a variant of the single scattering optical model.

5.3.3.1 Interpolation

By default, MARS Vision's DVR algorithm uses trilinear interpolation, as it offers a balance between quality and performance. In addition, it is natively supported by the GPU hardware platform that this algorithm is executed on.

Alternatively, the user is given a choice to enable an advanced interpolation scheme that uses basis splines, or B-splines [256]. This algorithm is known as cubic B-spline interpolation [95]. It samples a larger number of voxels (64, as opposed to 8 for trilinear interpolation), produces smoother surfaces, and reduces wood-grain artefacts. However, cubic B-spline interpolation is not used by default, as it is more computationally intensive than trilinear interpolation. The performance difference is further discussed in section 5.5.

5.3.4 Illumination

After the scattering location is determined, a ray is cast from the scattering point to one of the lights in the scene. The light is chosen randomly. The ray marches towards the light, samples the volume data at regular intervals, and uses the transfer functions to determine the opacity at each sampling point. The opacity is accumulated and used to determine whether the ray is able to reach the chosen light source.

If multiple volumes are present, then there are two options for simulating the light transfer. The first, called the "standard" lighting model, calculates it separately for each volume. This shown in Figure 5.12A: a ray that is cast from a particular volume towards a light source accumulates opacity by only sampling that volume. Other volumes have no effect on the lighting step (although all volumes are still sampled to determine the scattering position, as described in section 5.3.3). Results are shown on the left of Figure 5.13.

An alternative "realistic" lighting model attempts to correctly simulate the light transfer through *all* volumes of the dataset. A ray cast from any volume samples all other volumes to accumulate opacity. This process is shown in Figure 5.12B. The realistic lighting model increases the complexity of the illumination step: the standard model requires n lighting passes (one for each volume), while

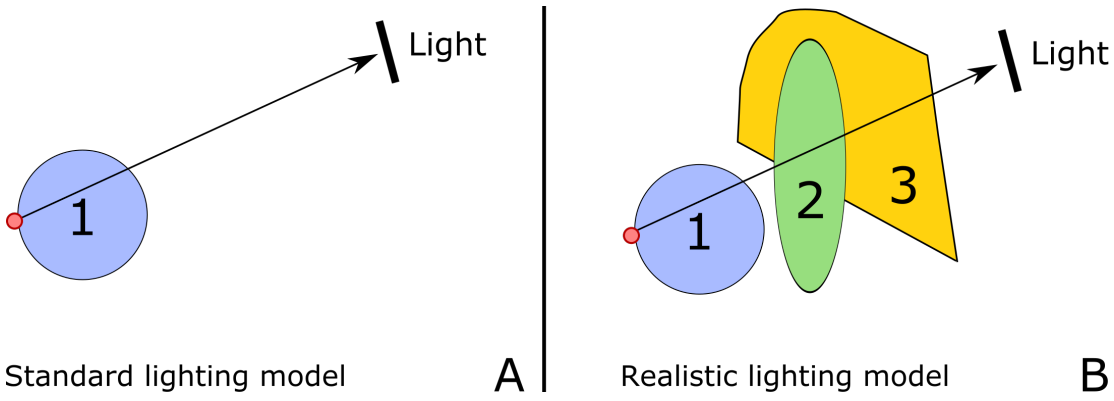


Figure 5.12: Standard (**A**) and realistic (**B**) lighting models in MARS Vision.

the realistic model requires n^2 lighting passes. This reduces the performance of MARS Vision's DVR algorithm when the realistic model is enabled. However, the frame rates remain interactive, as demonstrated in section 5.5.

This results in more physically correct illumination and shadows, as shown on the right side of Figure 5.13. This can improve the user's depth perception, as noted by Fuchs and Hauser [120] and Gribble and Parker [240].

Consider the example shown in Figure 5.13B: by default (left image), the calcium volume, shown in white, does not interact with the lipid volume, shown in beige. The light transfer simulation is carried out as though the lipid volume is not positioned between the calcium and the light source. The same applies to the lipid volume. However, the light source is positioned such that, if this was a real object, shadows should be cast by calcium onto lipid, and vice versa, as shown on the right.

Another example is shown in Figure 5.13C. The calcium (white) and water (red) volumes overlap in the highlighted region. Here, the surface geometry is complex, and the standard lighting model does not provide sufficient detail to clearly display the spatial relationship between these two volumes. The realistic model correctly simulates the light transfer by sampling all volumes, which provides additional depth cues. Because of that, the spatial relationship between the features of the calcium and water volumes is shown much more clearly.

The best example of the advantages of the realistic lighting mode is shown in Figure 5.13D. The volume of iodine (blue/purple/red) is positioned above the volume of calcium (white), but the standard lighting mode does not clearly show this relationship. It provides misleading and confusing depth cues because the

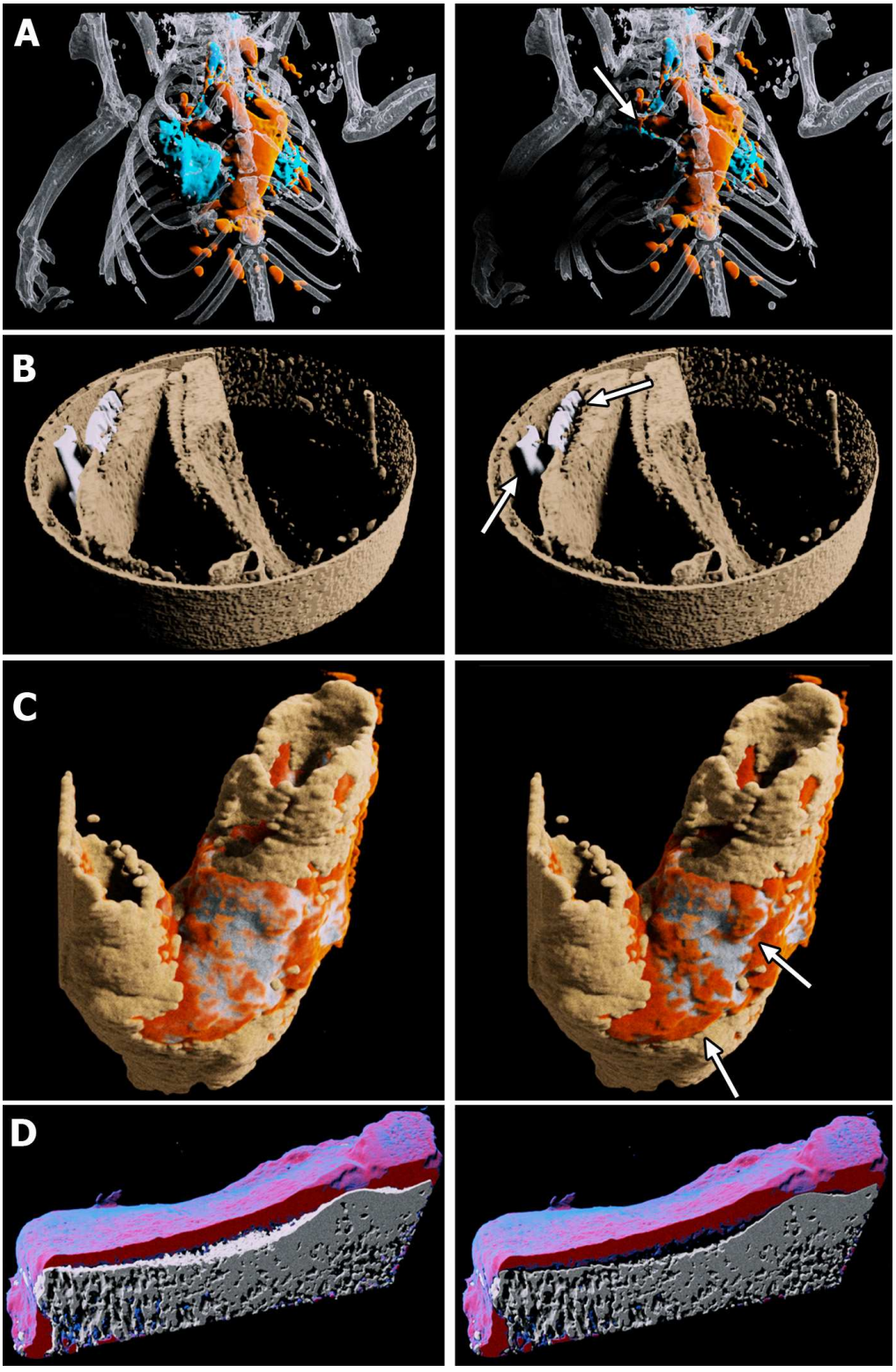


Figure 5.13: Visualisation of three spectral CT datasets with the standard lighting model (left) and the realistic lighting model (right). A single directional light source is used to clearly show the effects. Areas where the differences are most obvious are marked by white arrows. **A**: Mouse12 (section 9.6.1). **B**: Meat112 (section 9.3). **C**: Plaque108 (section 9.2). **D**: Knee Cartilage (section 9.4.1).

light, which is coming from the direction marked by the orange arrow, appears to penetrate the iodine volume and illuminate the top surface of the calcium volume. This should not occur, as the iodine volume is opaque and should not let any light through. The realistic lighting mode corrects this error and helps the user analyse the dataset by clearly displaying the correct spatial relationship between the iodine and calcium volumes.

Realistic illumination may not always be a desired effect, as it may decrease the visibility of some parts of the image. For example, consider the Mouse12 dataset (Figure 5.13A): the barium volume (light blue) is visible with the standard model. However, if the realistic model is used, then the barium volume is mostly hidden by the shadows cast by the calcium (white) and iodine (orange) volumes.

Therefore, it may be advantageous to use a simpler lighting model for the purposes of visualisation. This situation is an example of how non-photorealistic rendering techniques may bring out the features that are not clearly shown if realistic rendering techniques are used. Usually, an approximation provided by the standard model is sufficient. In the cases where the surface geometry is complex, or features from multiple volumes are overlapping, the realistic lighting model may be a better option, as it may help the user better understand the spatial relationships between the features of multiple volumes.

By default, the standard model is used, as there is a smaller chance of ROIs being obscured by shadows cast by other structures. However, this varies with each dataset, so the user is always given the option to switch between the two illumination modes.

5.3.5 Colour blending, Monte-Carlo integration and output

The illumination step generates a colour value and an opacity percentage for each volume. The colour is multiplied by the associated opacity, and all colours are additively blended together in the CIE XYZ colour space. The aim is to calculate a weighted average, where the contribution of each colour is dependent on the opacity associated with it. This is done by calculating the sum of the opacities of all volumes, which is used to correct the brightness of the image after the colour blending step:

$$\text{Blended colour} = \frac{\sum_{i=1}^{n} C_{\text{i}} \times O_{\text{i}}}{\sum_{i=1}^{n} O_{\text{i}}} \tag{5.1}$$

where C_i and O_i are, respectively, the calculated colour and opacity for the i-th volume. This step was not required in Exposure Render, which did not support multiple volumes.

The frame estimate generated during an iteration is blended with the sum over multiple iterations (the running estimate) according to the algorithm originally implemented in Exposure Render. The colour (in the CIE XYZ colour space) of each pixel in the running estimate buffer is computed as follows:

$$\text{Running estimate} = \text{R} + \frac{\text{F-R}}{\text{number of iterations}} \tag{5.2}$$

where R is the old value of running estimate (before the beginning of the current iteration) and F is the value of the frame estimate.

After Monte-Carlo integration, the running estimate buffer is converted into the RGB colour space before being displayed. MARS Vision optimises this process by using the CUDA/OpenGL interoperability functions to directly draw the buffer stored on the GPU and avoid copying it between VRAM and RAM.

5.3.5.1 Colour replacement mode

The use of colour to visually separate multiple volumes does not always produce a clear result. That is, it is not always clear which volumes have contributed to the final colour of the pixel displayed on the screen. This situation is often observed when several volumes overlap, and affects both 2D (as discussed in section 4.3.1.4), and 3D techniques. This section explains the implementation of the colour replacement mode in MARS Vision's DVR algorithm, while section 5.6.4.1 describes colour replacement during 2D slice visualisation.

Figure 5.14 illustrates the problems associated with colour blending. In this example, an energy volume is shown in grey to provide the context, while the two identified materials are shown in colour. Using the default algorithm (B) leads to the blending of colours assigned to the barium (cyan/blue), iodine (orange/red), and energy volume channels.

Blending of this kind may be a desired effect, but it may also interfere with

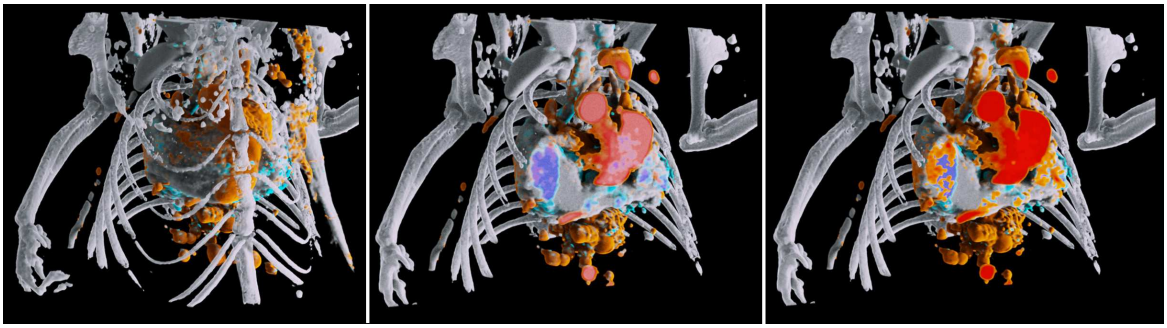


Figure 5.14: The colour replacement mode of MARS Vision's DVR algorithm. **A**: DVR of the Mouse12 dataset (section 9.6.1). **B**: The clipping plane set to reveal the detail inside the chest of the mouse, where all three channels overlap. **C**: Colour replacement by the iodine volume, which preserves the colour gradient for this particular material.

the visualisation of a colour gradient for a particular material. This is the reason for implementing the *colour replacement* mode.

This mode does not perform blending (section 5.3.5) if the colour calculated for the replacement volume is associated with a non-zero opacity, as classified by the transfer function. Instead, this colour is set as the final pixel colour, and the colours calculated for all other volumes are discarded. Therefore, the colour replacement mode can be thought of as a non-photorealistic rendering technique that avoids any colour distortion caused by blending and preserves the colour gradient assigned to a single channel. The results are shown in Figure 5.14C.

5.3.6 Clipping

Clipping (also referred to as slicing or cutting) is the process of hiding a part of a volumetric dataset during visualisation. The source volume data is not altered. Clipping is usually carried out by moving a virtual clipping plane through the dataset. All data on one side of the plane is hidden, while the data on the other side is rendered normally.

Clipping planes allow the user to look inside the dataset and remove the structures that may be occluding regions of interest. Clipping planes may be axis-aligned, or oblique (arbitrary orientation). MARS Vision supports both types, as shown in Figure 5.15.

Finally, a clipping box (Figure 5.15D) has also been implemented. It can be used to retain more context by restricting clipping to a particular region. In contrast, the clipping planes remove all data on one side of the plane. For example,

the structure inside the lungs can be displayed by positioning a clipping plane inside the chest (Figure 5.15B). However, this loses some context, which can be preserved through the use of a clipping box (Figure 5.15D).

The clipping box is placed independently of the clipping planes and allows the user to cut out an axis-aligned cuboid inside the dataset. The clipping box is placed by clicking on a visible surface inside the volume, in which case the depth buffer (section 5.4) is used to translate the mouse click coordinates into volume coordinates. The size of the clipping box can be adjusted by moving three sliders that control its x, y and z dimensions.

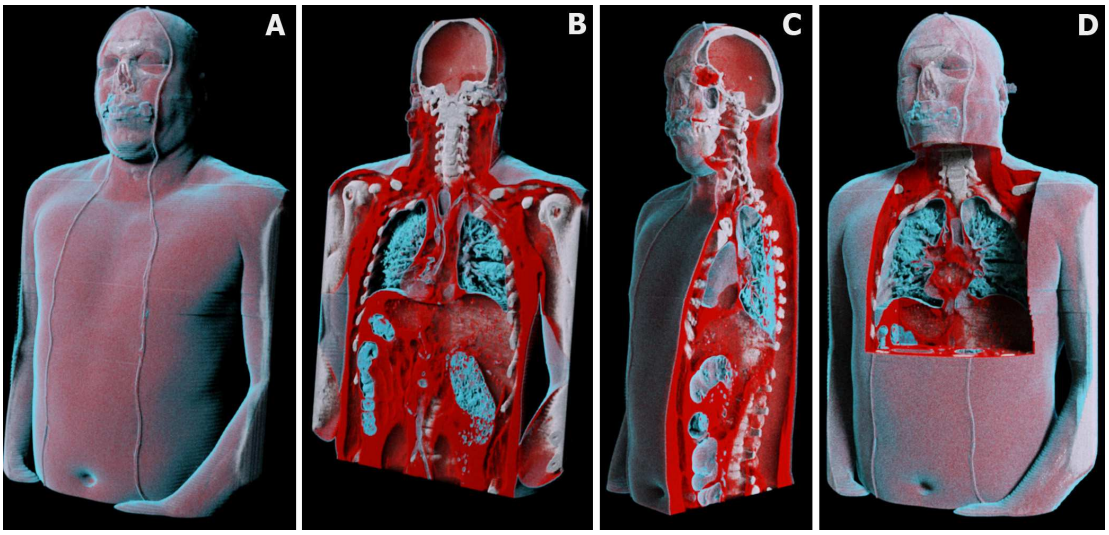


Figure 5.15: Clipping planes and a clipping box implemented in MARS Vision. **A**: No clipping. **B-C**: clipping planes. **D**: clipping box.

5.3.7 Summary

MARS Vision's DVR algorithm is a modified version of the algorithm originally implemented in Exposure Render. The algorithm has been extended to render multiple volumes simultaneously, provide a realistic lighting model, support clipping planes and a clipping box, and include an alternative interpolation scheme based on cubic B-splines.

5.4 Interaction using a depth buffer

This section discusses the data structure needed for interacting with 3D DVR visualisation in MARS Vision. Most visualisation applications implement a GUI based on the standard mouse and keyboard input devices. This is the case for general-purpose image processing software such as ImageJ [67], specialised medical and scientific visualisation software such as 3D Slicer [68] and MITK [69], and various DICOM viewers [78,80,81,231]. Finally, both custom spectral CT visualisation applications created by the MARS team (MARSCTExplorer and MARSCUDAVR, as described in section 4.3.2) also use mouse and keyboard input [74, 129].

In a typical 3D visualisation application, the mouse is normally used to rotate, pan, and zoom the virtual camera in the 3D scene. MARS Vision implements this functionality, and extends it by adding direct mouse-based interaction with volume data.

Rotation, panning, and zoom can be achieved with, respectively, the left and middle mouse buttons, and the mouse wheel. This leaves the right mouse button, which is used to carry out tasks such as voxel selection and measurement (section 8.1), voxel editing (section 8.5), and annotation painting (section 8.1.4) by clicking on visible surfaces in the 3D view. This section describes the implementation of the data structure that allows this type of interaction to take place.

A standard DVR algorithm renders an image, but provides no auxiliary information needed to interact with the visualisation. Interaction requires a 2D mouse click coordinate to be translated into a 3D coordinate inside the volume.

MARS Vision accomplishes this by maintaining a buffer that maps each image canvas pixel to a 3D voxel coordinate. For convenience, this buffer is referred to as a "depth buffer". A standard depth buffer (also known as a z-buffer), is an essential part of the standard modern graphics rendering pipeline. It provides a way of resolving the visibility conflicts that occur when one object in a 3D scene occludes another [257].

The depth buffer is automatically maintained if the standard graphics rendering pipeline is used. However, DVR algorithms do not use this pipeline and thus have no access to the standard depth buffer.

MARS Vision implements a custom depth buffer. The buffer is stored as a 2D array on the GPU, and updated during each iteration of the DVR algorithm. It allows the user to naturally interact with the 3D visualisation by selecting points

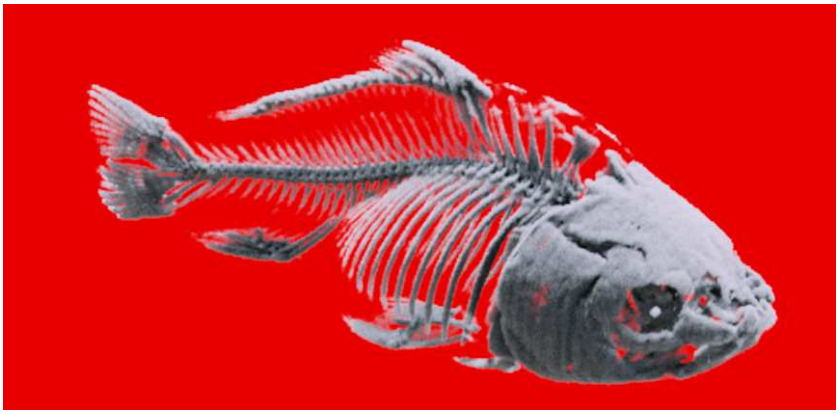


Figure 5.16: Visualisation of the Carp dataset. The image pixels associated with invalid depth buffer positions are shown in red. All rays cast from a pixel must fail to find a valid sampling point in order for that pixel to be marked as invalid.

on visible surfaces. The regions made transparent by the transfer function will be associated with invalid depth buffer values, as shown in Figure 5.16.

Definitions:
Valid point: a pixel inside the image canvas associated with a valid voxel coordinate. This means that at least one sampling ray cast from this pixel has found a valid scattering position inside the dataset.

Invalid point: a pixel inside the image canvas that is associated with an invalid voxel coordinate. This means that all sampling rays cast from this pixel have failed to find a valid scattering position inside the dataset.

5.4.1 Initialisation, maintenance and use of the depth buffer

The tasks of initialising and maintaining the depth buffer are integrated into the DVR algorithm described in section 5.3. Initially, the depth buffer location for each image pixel is set to infinity. Any change to transfer function parameters or camera position restarts the rendering process and resets the depth buffer.

During an iteration of MARS Vision's DVR algorithm, each ray finds a scattering location inside the dataset as part of the stochastic raycasting step (section 5.3.3). Updating the depth buffer involves an additional comparison performed after this point has been found.

If the point is valid (inside the dataset's bounding box), then it is compared to the location stored in the depth buffer. If the new scattering point is closer to the virtual camera's position than the stored point, then the depth buffer is updated with the new position.

The depth buffer is stored on the GPU. Therefore, before it can be used, it must first be copied from the GPU's VRAM into the system RAM. However, once copied, the depth buffer can be accessed directly, as its dimensions are the same as the dimensions of the rendered image. The *(x, y)* coordinate of the mouse click can be directly mapped to a particular element of the depth buffer. These coordinates stored in the depth buffer can be used to measure the voxel's value, initiate segmentation, or place annotations. The coordinates are accurate to the nearest voxel.

5.4.2 Convergence of the depth buffer

Kroes et al. have demonstrated that Exposure Render's Monte-Carlo estimate converges rapidly, usually in under a second on a high-end GPU [103]. This section considers convergence for the purpose of interaction. It determines how soon the user the can query the depth buffer and receive a position that is not going to change substantially as subsequent iterations are executed.

To test this, several images were rendered using different datasets (Plaque77, section 9.2.1); Mouse12, section 9.6.1; Meat1127, section 9.3), transfer functions and opacity settings. The images are shown in Figure 5.17.

Two different convergence types are tested:

1. How soon all invalid image pixels are found. Figure 5.18 shows that, if the depth buffer value is invalid after several iterations, then it is likely to remain invalid after 100 iterations (the value of 100 has been chosen as the limit because, after that point, the improvement in image quality from each subsequent iteration is difficult to perceive visually). This means that invalid image pixels are found rapidly.

2. How soon a reasonably accurate depth buffer value is found for each valid image pixel. Figure 5.19 shows that, if the depth buffer position is valid, then it is likely to be very close to the value after 100 iterations. This means that each valid pixel is quickly associated with a certain depth buffer value,

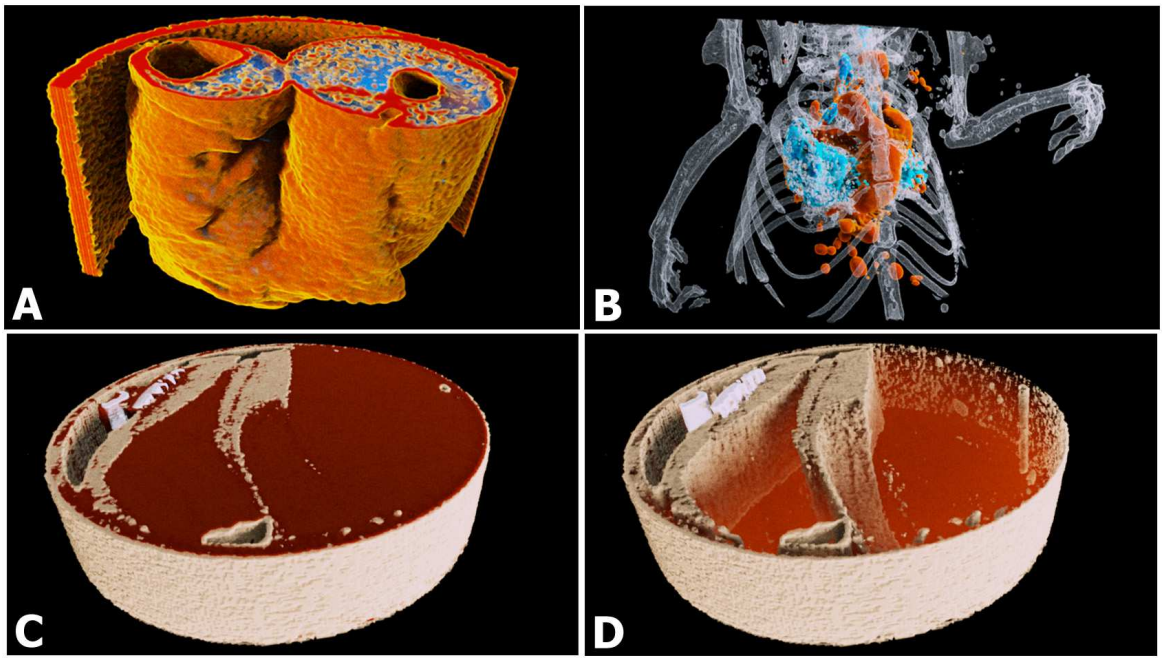


Figure 5.17: Images rendered to test the convergence of the depth buffer. **A**: Plaque77 (section 9.2.1). **B**: Mouse12 (section 9.6.1). **C**: Meat1127 (section 9.3.1), high opacity settings. **D**: Meat1127, low opacity for the water channel. All images have been rendered at the resolution of 1280×720 pixels. Rendering has been stopped at 100 iterations.

which is not going to change significantly as rendering continues, and more iterations are executed.

In addition, these figures demonstrate that the speed of convergence is dependent on the opacity of the volumes. When high opacity settings are used (Figure 5.17C), then convergence is extremely rapid, with around 5 iterations being required to generate positions within 1% of the final position. However, when the opacity is decreased (Figure 5.17D), the convergence is much slower. The same 1% distance is now reached at around 20 iterations.

Figure 5.17B shows another situation where low opacity leads to slow depth buffer convergence. In general, the lower the opacity, the slower the depth buffer will converge. Nevertheless, reasonably accurate depth buffer positions will be calculated quickly, fewer than 30 iterations are required in most cases, as shown in Figure 5.19. This number of iterations is usually reached in under a than a second, as explained in the next section.

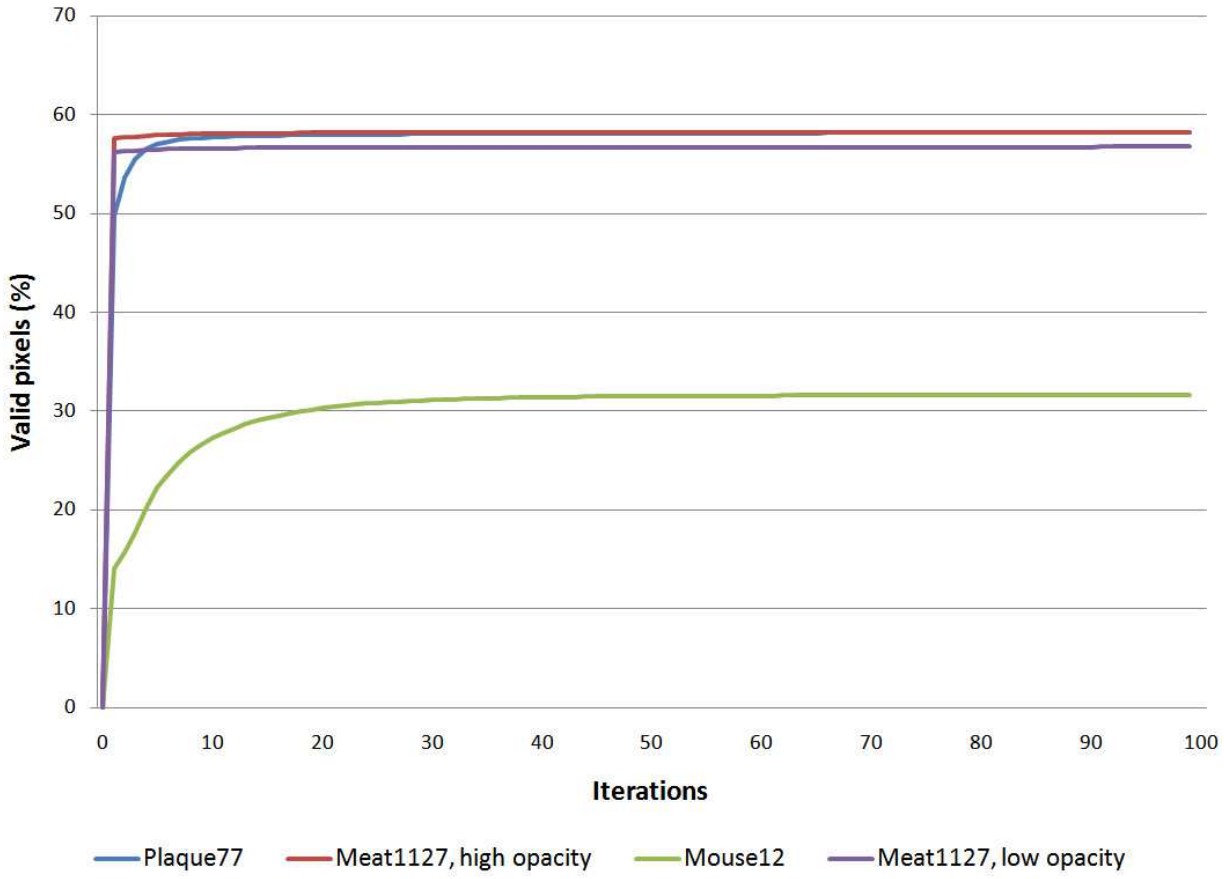


Figure 5.18: The percentage of valid pixels in the depth buffers generated for the four images shown in Figure 5.17.

5.4.3 Summary

MARS Vision implements a custom depth buffer that associates each pixel of the rendered image with a voxel coordinate that corresponds to the closest valid sampling point inside the dataset. The depth buffer is updated during each iteration of the volume rendering algorithm and allows the user to directly interact with volume data by selecting points on visible surfaces.

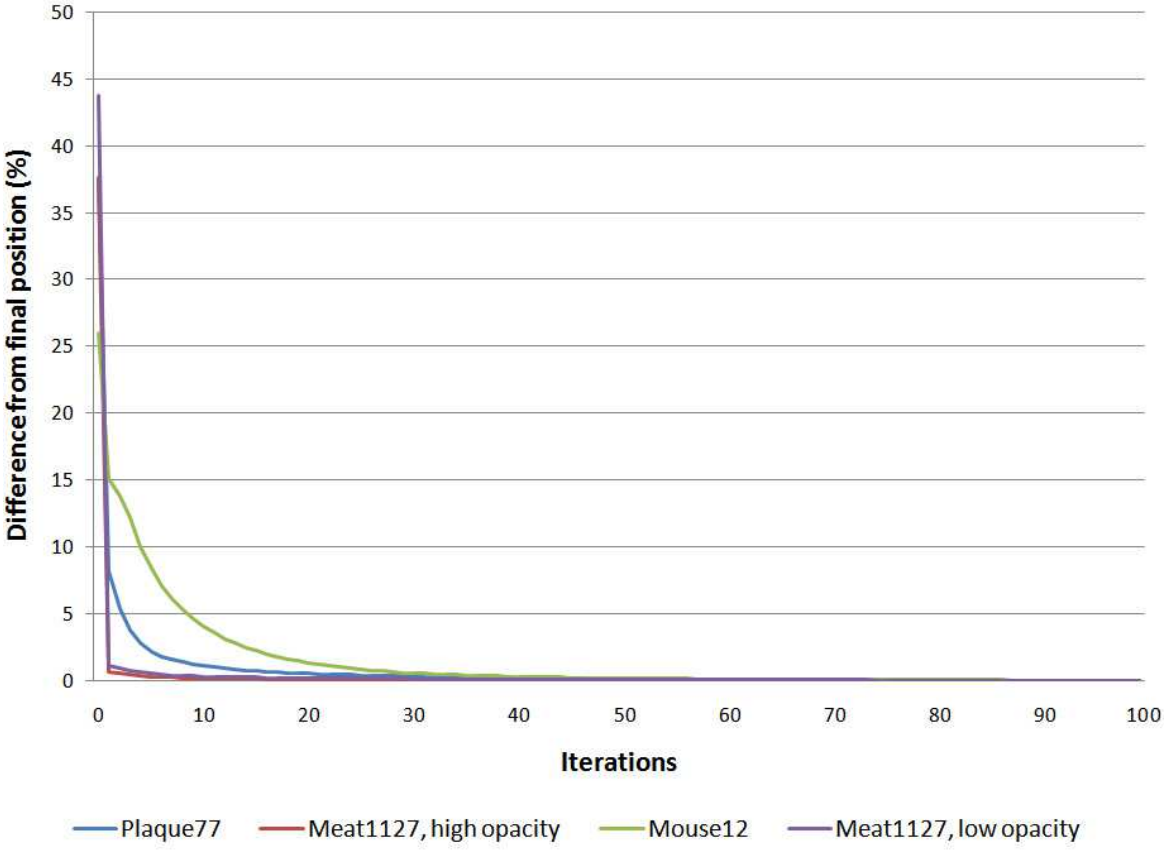


Figure 5.19: Convergence of the depth buffers for the four images shown in Figure 5.17. The percentage difference from the final depth buffer position has been found by: (a) calculating the distance between the position after 100 iterations and the position after each iteration, and (b) scaling the distance by the dimensions of the dataset to convert it to a percentage.

5.5 Optimisation and performance

The DVR algorithm originally implemented in Exposure Render does not require optimisation to maintain interactive frame rates when rendering datasets comprising a single volume. This includes conventional CT and MRI datasets. However, simultaneous multi-volume rendering is a requirement for the MARS project, as MARS spectral CT datasets always include multiple energy and/or material volumes. In this case, optimisations are required to achieve interactive performance.

Before evaluating the performance of MARS Vision's DVR algorithm, it is important to establish the number of iterations required to obtain a reasonable-

quality image. As shown in Figure 5.6, the major structures in the image are already visible after the first 1-5 iterations, the high-frequency noise in the image is reduced after around 20 iterations, and there is little change in visual quality after 50 iterations. There is no upper limit to the number of iterations that can be rendered.

During interaction (transfer function adjustment, light placement, camera manipulation), a rendering speed of over 10 iterations per second is generally sufficient. Once interaction stops, the image quality will progressively improve. Therefore, it may take over a second (or up to 5-10 seconds for larger images) to generate a high-quality image, but an estimate showing all major features will be calculated very quickly.

The performance of MARS Vision's DVR algorithm is inversely proportional to the number of volumes: an iteration finds a scattering location in each volume, and then calculates the illumination for each volume. Other factors affecting the performance include the dimensions of the rendered image, the sampling step size, transfer function and opacity settings, capabilities of GPU hardware, and so on.

Therefore, a range of datasets and transfer function settings should be used to test the performance, as shown in Figure 5.20. Table 5.1 shows the performance of MARS Vision's DVR algorithm when rendering these datasets using the GeForce GTX670 GPU.

Table 5.1: Average frame rates (iterations per second) for seven MARS spectral CT datasets shown in Figure 5.20.

Dataset	Dimensions, volumes	Unoptimised, motion and stationary	Optimised, motion	Optimised, stationary
Mouse12	$338 \times 440 \times 257, 4$	7.20	37.95	7.41
Spectral Phantom	$246 \times 246 \times 16, 5$	30.67	93.81	31.37
Meat1127	$436 \times 436 \times 126, 3$	28.35	102.93	38.41
Mouse0	$414 \times 414 \times 451, 4$	10.22	44.98	13.90
Knee Cartilage	$544 \times 460 \times 587, 2$	19.36	96.33	21.85
Plaque108	$384 \times 384 \times 288, 3$	13.63	95.17	30.64
Plaque72	$583 \times 583 \times 99, 4$	12.70	87.75	30.57

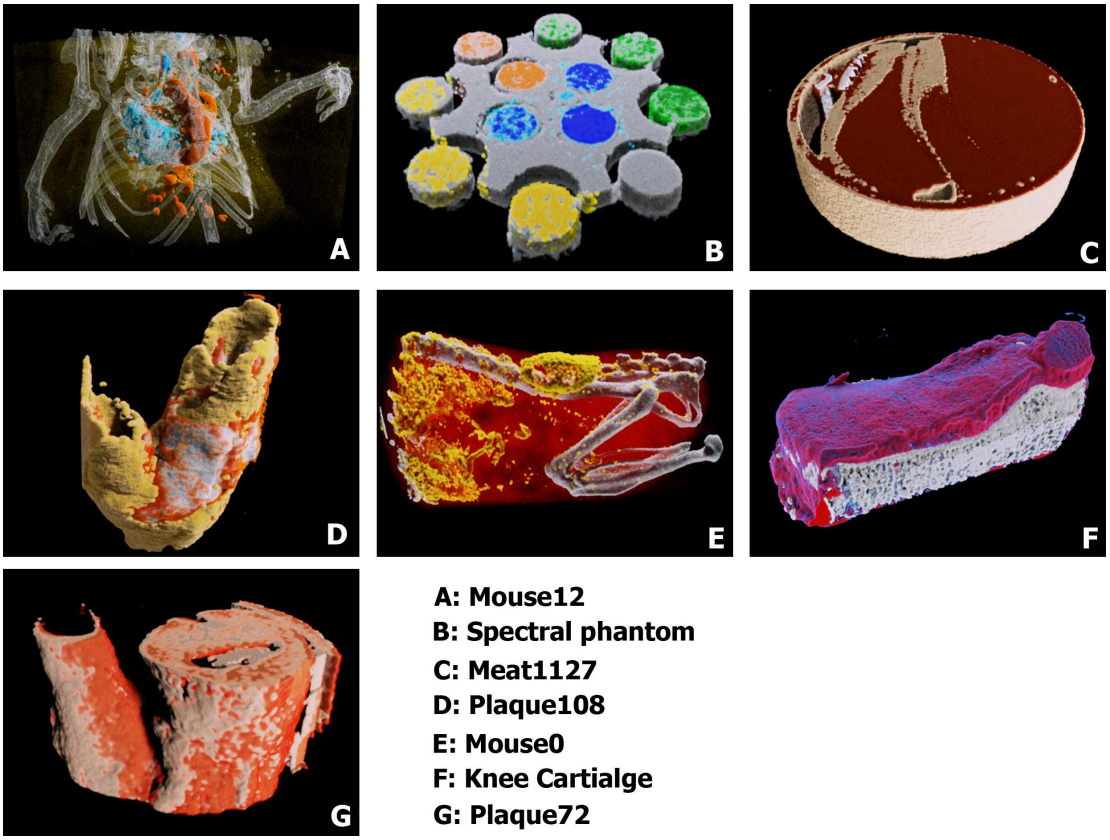


Figure 5.20: Seven MARS spectral CT datasets rendered at 800×600. The frame rates for these datasets are shown in Table 5.1.

Without optimisation, this algorithm is able to sustain an interactive frame rate when rendering MARS spectral CT datasets. However, the frame rate can be improved substantially if standard DVR optimisation techniques are implemented.

Optimisation techniques for DVR include dynamic image quality adjustment [258], early ray termination [104], empty space skipping (often based on an octree [259] or a k-d tree [260]), empty-space leaping [105, 261], and adaptive sampling [262]. The optimisations implemented in MARS Vision are described below.

5.5.1 Dynamic image quality adjustment

Dynamic image quality adjustment is a basic, but effective optimisation technique. Generally, it involves increasing the sampling step size, or turning off non-essential image enhancements [258]. This leads to the loss of fine detail during camera motion. However, during interaction, the fluidity of motion is usually more important

than the image quality [62].

During interaction (such as camera motion, or adjustment of the transfer function), MARS Vision changes sampling rate and the image resolution:

- The sampling rate determines the distance a ray advances after taking a sample. This affects both the scattering point selection step, and the illumination calculation step. A high sampling rate results in smoother surfaces and fewer sampling artefacts, while a lower sampling rate increases the performance.

  MARS Vision uses a linear scale that ranges from 1 to 10, where 1 corresponds to the highest possible sampling rate, and 10 corresponds to the lowest possible sampling rate. This means that, after it samples the dataset, a ray can advance by a distance equivalent to the size of 1-10 voxels.

  When rendering restarts, the sampling rate is set to 10, and is then progressively returned to its original setting over several iterations. This is done according to the following formula:

$$\mathrm{R} = \max(\mathrm{R},\ 10 - (\mathrm{N} \times 2.5)) \tag{5.3}$$

  where R is the sampling rate, and N is the number of iterations. This algorithm decreases the sampling rate parameter by 2.5 after each iteration until the value set by the user is reached. For example, a target of 3.0 will be reached after 3 iterations (first iteration: 10; second iteration: 7.5; third iteration: 5.0; fourth iteration: 3.0).

- When rendering restarts, the resolution of the image is reduced by a factor of two in each dimension. After several iterations, the full resolution is restored. A CUDA kernel is used to upscale the contents of the running estimate and the depth buffers. The user is given the option to enable or disable this optimisation. In addition, the user is able to set the number of iterations that must elapse before the transition to the full resolution takes place.

5.5.2 Empty space skipping

Empty-space skipping (ESS) is known to be an effective acceleration technique for single-volume DVR [263]. ESS allows the DVR algorithm to avoid unnecessary sampling in empty (as determined by the transfer function) regions of the volume.

A naïve ray marching DVR algorithm is a brute-force technique. It casts hundreds of thousands of rays into the volume and samples the volume data millions of times, but only a fraction of these samples contributes to the final image displayed to the user. Most samples are made inside regions of empty space, and can be skipped without affecting the image quality.

The fraction of non-transparent samples is dependent on the volume and the transfer function settings. For instance, Levoy finds that this fraction ranges from 9 to 27% [94]. However, a naïve ray marching DVR algorithm has no prior information about the transparency of the volume. Therefore, a ray, marching through the volume, must sample at every step, even though most samples are likely to contribute nothing to the final pixel colour [104].

The key to accelerating rendering is reducing sampling in these "empty" regions. To do this, MARS Vision uses a space-partitioning data structure, called an *octree* [259], to classify blocks of voxels. This enables a more intelligent approach to sampling. Prior research has found that using octree-based ESS increases the performance of DVR by 200-600%, depending on the dataset and the transfer function settings [74, 263].

In MARS Vision, the octree is maintained as a 3D array stored on the GPU. It is a downsampled representation of the dataset, as shown in Figure 5.21. Every element of the octree stores the average opacity of a block of 4^3 voxels in all *visible* volumes. This means that, once hidden, a volume is ignored by the octree generation algorithm.

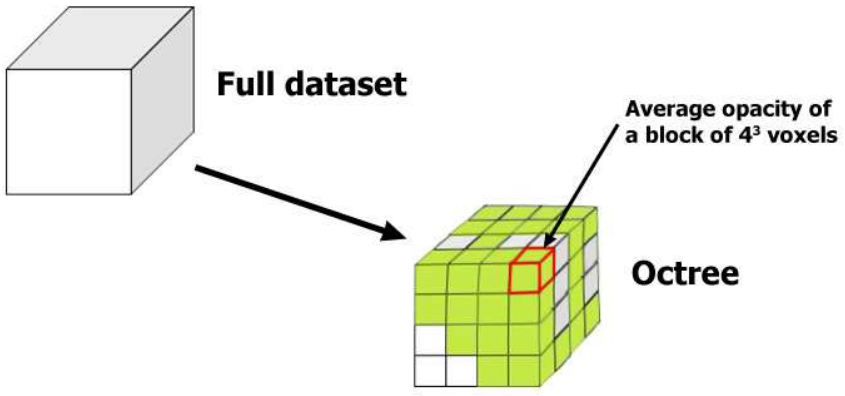


Figure 5.21: MARS Vision's octree, which is a downsampled version of the original dataset. The octree is stored as a 3D texture on the GPU.

The octree generation procedure is shown in Algorithm 2. For clarity, this algorithm omits boundary checking and assumes that the coordinates of each 4^3 block of voxels are known.

Algorithm 2 Calculation of the average opacity of a block of 4^3 voxels during octree generation.

Data: n volumes $V_{1...n}$; n transfer functions $T_{1...n}$
Result: opacity for the block O

initialization:
$O \leftarrow 0$
loop through all volumes, calculate the opacity of a block in each volume, and accumulate the opacity:
for $i \leftarrow 1$ **to** n **do**
    retrieve the 4^3 block of voxels from V_i
    calculate opacity of each voxel in the block using T_i
    accumulate all opacity values to find the total opacity of the block
    O += total opacity of the block $\div$ 4^3
end

save O as a percentage

Figure 5.22 displays the octree generation times for six MARS spectral CT dataset (Mouse0 (section 9.6.2), Mouse12 (section 9.6.1), Meat1127 (section 9.3), Plaque72 (section 9.2.1), Knee Cartilage (section 9.4), and Plaque108 (section 9.2.2). It demonstrates that the octree generation time scales linearly with the number of voxels. For example, an octree based on 2 volumes from the Mouse0 dataset is generated in around 25 ms, while an octree based on all 5 volumes is generated in around 55 ms. Most current MARS datasets are processed in less than 50 ms. Therefore, the creation of the octree can take place during visualisation and has little effect on the performance.

The scattering point determination algorithm (section 5.3.3) and the lighting calculation algorithm (section 5.3.4) both benefit from ESS. Before sampling any volume, these algorithms use the octree to check the opacity at the sampling point. If the opacity is below the threshold, no sample is taken, and the ray advances.

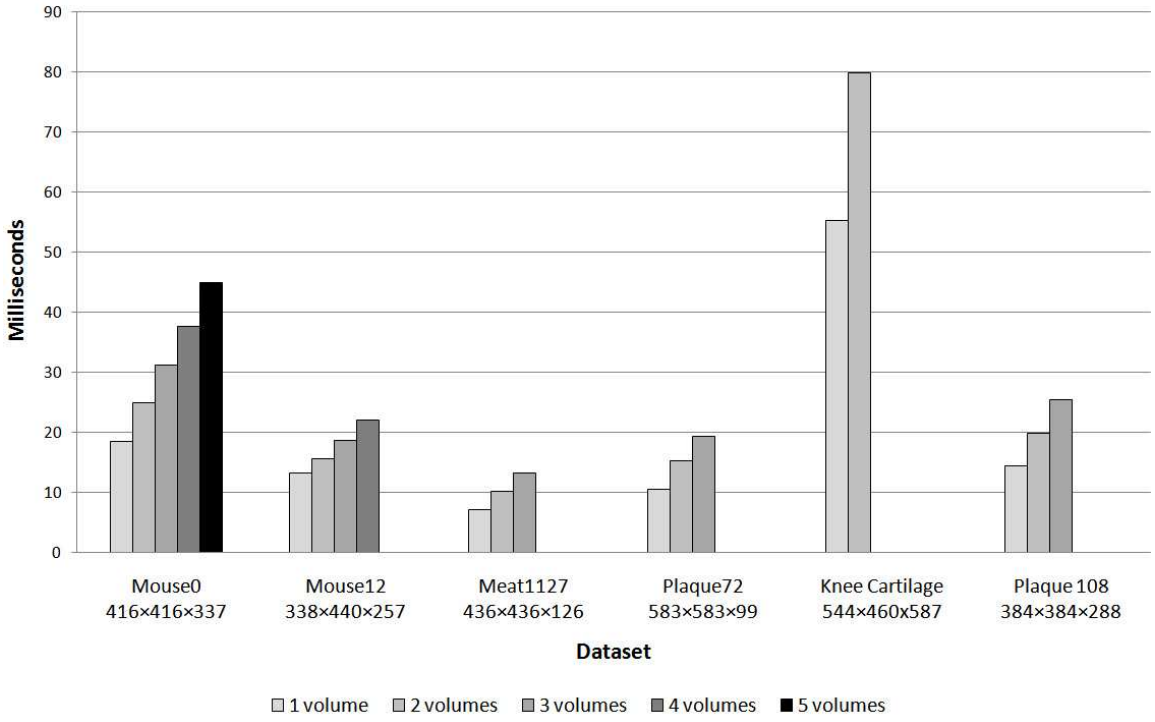


Figure 5.22: Octree generation times for six MARS spectral CT datasets. The dimensions of each volume in the dataset are shown below the dataset name.

5.5.3 *Effects of the realistic lighting model*

Table 5.2 shows that, in most cases, the realistic lighting model does not significantly affect the rendering speed. Therefore, the choice between the standard and the realistic lighting models can be left to the user, as both are useful in different circumstances (see section 5.3.4).

5.5.4 *Effects of cubic B-spline interpolation*

The last issue to consider is the effect of the cubic B-spline interpolation algorithm on the performance. As mentioned in section 5.3.3.1, it is an optional interpolation technique that samples 64 neighbouring voxels around the sampling location and calculates the contribution of each voxel by fitting a cubic B-spline. Standard trilinear interpolation only samples 8 voxels, so we can expect the performance to decrease substantially.

However, Table 5.3 shows that this is not the case: the performance only drops by around 15-50%. This is because:

Table 5.2: Effect of the realistic lighting model on the performance of MARS Vision. The same datasets and rendering parameters as those described in Table 5.1 are used. The camera is stationary and all optimisations are applied. Measurements are in iterations per second.

Dataset	Standard model	Realistic model	Performance decrease (%)
Mouse12	7.41	4.64	37.38
Spectral Phantom	31.37	28.70	8.51
Meat1127	38.41	35.06	8.72
Mouse0	13.90	12.11	12.88
Knee Cartilage	21.85	18.66	14.60
Plaque108	30.64	27.59	9.95
Plaque72	30.57	24.45	20.02

- Interpolation is only performed during the scattering point selection step (section 5.3.3), and the first part of the illumination step, which attempts to reach a light source by casting a ray from the chosen scattering point (section 5.3.4). Other algorithms, such as shading (the second part of the illumination step), blending, denoising, and Monte-Carlo integration, are unaffected.

- The GPU performs automatic texture caching by storing commonly-accessed

Table 5.3: Effect of cubic B-spline interpolation on the performance of MARS Vision. The same datasets and rendering parameters as those described in Table 5.1 are used. The camera is stationary and all optimisations are applied. Measurements are in iterations per second.

Dataset	Trilinear interpolation	Cubic B-spline interpolation	Performance decrease (%)
Mouse12	7.41	3.28	55.72
Spectral Phantom	31.37	23.47	25.19
Meat1127	38.41	32.70	14.86
Mouse0	13.90	7.48	46.17
Knee Cartilage	21.85	14.71	32.92
Plaque108	30.64	25.24	17.61
Plaque72	30.57	21.04	31.17

parts of 2D and 3D textures in low-latency cache memory [149]. This means that, once one interpolation has been performed, subsequent interpolations in the same region of the dataset are likely to be accelerated by the re-use of cached data.

Therefore, cubic B-spline interpolation is computationally intensive, but can still be used for interactive work, such as adjusting the camera position, or changing the transfer function. In practice, trilinear interpolation is usually sufficient for these tasks; cubic B-spline interpolation can be enabled once the appropriate transfer function, lighting, and camera parameters have been found.

5.5.5 Summary

MARS Vision's DVR algorithm is accelerated using several optimisation techniques that ensure interactive performance on commodity GPU hardware. During interaction, the image quality settings are reduced, and the image is rendered at a lower resolution. In addition, the algorithm incorporates octree-based empty-space skipping.

These optimisations do not lead to a substantial increase in frame rate when the camera is stationary. However, they vastly improve the performance during interaction, when the fluidity of motion is most needed. Overall, as shown in Table 5.1, performance during motion is increased by 300-400% in comparison to the unoptimised version.

5.6 2D visualisation

This section describes the implementation of various 2D visualisation modes and the algorithm for performing 2D slice data fusion based on the same transfer function settings that are used for DVR. Other features usually associated with 2D visualisation (such as annotation, segmentation, and measurement) are described in Chapter 8.

MARS Vision supports four different slice view orientations: the three standard axis-aligned views (sagittal, axial and coronal), and an oblique slicing plane. However, the functionality of each view, and the GUI presented to the user, are identical. All volume data is stored in arrays in RAM, so multiplanar reconstruction (creating a series of slices along a different set of axes [20]) is obtained by

simply retrieving the appropriate voxels, forming a slice, and displaying it on the screen. An example is shown in Figure 5.23.

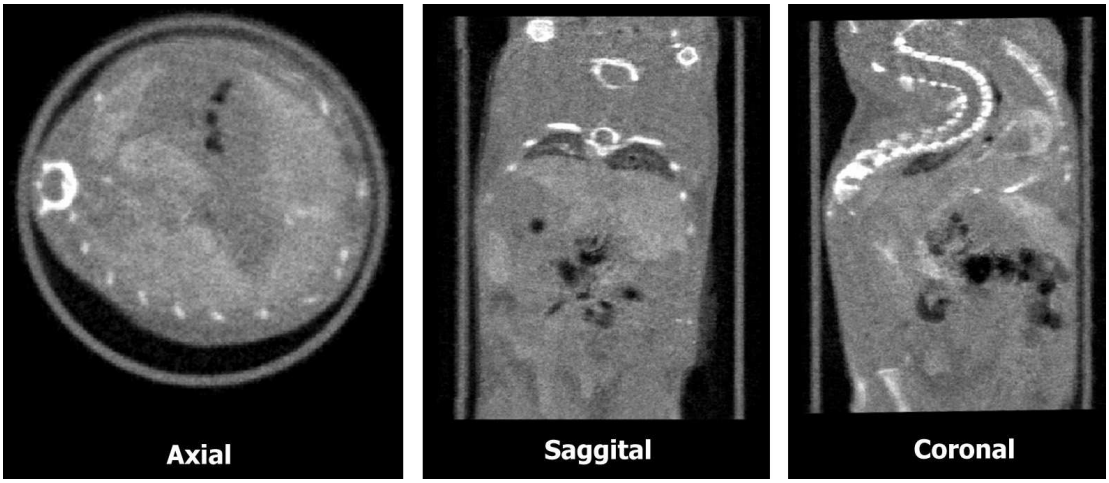


Figure 5.23: Multiplanar reconstruction of an energy volume of the Mouse0 dataset (section 9.6.2).

5.6.1 GUI

This section uses the term "slice view widget" to refer to the complete set of GUI elements for visualising a slice of a volume and navigating through the slices. Figure 5.24 shows the GUI of a slice view widget. It supports several tasks:

- Navigation through slices of the selected volume in a particular direction. The three directions are: sagittal (along the the x axis), coronal (along the y axis), and axial (along the z axis). As described in Chapter 4, this is currently the most common method of visualising spectral CT datasets.

- Switching between volumes. This is an essential requirement for spectral CT data visualisation, as described in section 4.4.1. A single slice view widget can be used to sequentially visualise slices of different volumes of a spectral CT dataset.

- Fusing the slices of multiple volumes. This is an additional requirement for spectral CT visualisation. The need for 2D data fusion has been explained in section 4.4.2.

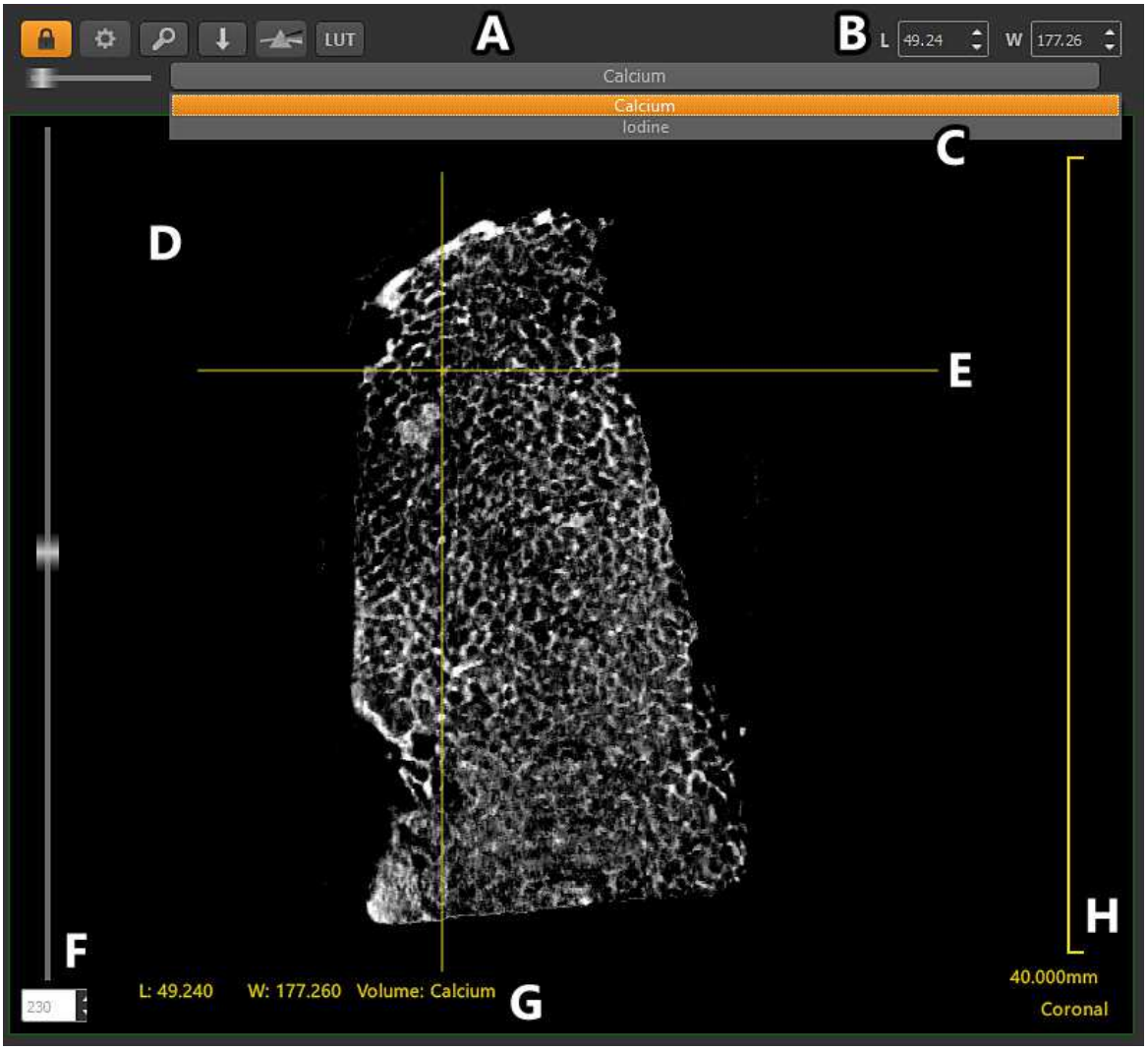


Figure 5.24: The GUI of a slice view widget. **A**: Main toolbar containing the buttons for switching between the three visualisation modes. **B**: Window and level settings. **C**: Volume selection combo box. **D**: The main image canvas. **E**: Crosshair showing the selected pixel and the synchronisation lines showing the indices of slices selected in other slices view widgets. **F**: the slider and spinner for selecting the slice index (alternatively, scrolling can be done with the mouse wheel). **G** and **H**: various annotations and information, such as the name of the volume and the scale bar.

Slice view widgets support three visualisation modes: standard, look-up table, and a mode that fuses slice data based on transfer functions (referred to as "spectral mode"). Each mode displays the slice data using different settings. Other features, such as navigation through slices, or ROI measurement, are unaffected by the choice of the mode.

5.6.2 Standard mode

The standard mode implements the traditional 2D slice visualisation technique based on a greyscale colour scheme controlled by window and level settings (section 2.2.1). The user is able to switch between the volumes using a combo box (Figure 5.24C).

This mode is needed for studying individual energy or material volumes, as shown in Figure 5.24, and for compatibility with other data types, for example, conventional CT or MRI datasets.

5.6.3 Look-up tables

The look-up tables (LUTs) implemented in MARS Vision are 1D texture maps, which can also be thought of as pre-made transfer functions that provide a simple way of assigning colour to volume data. LUTs are an intermediate step between the basic greyscale colour scheme generated using window and level settings, and the arbitrary mapping of colour with a transfer function editor.

The reason LUTs may in some cases be preferable is that transfer functions may be difficult and time-consuming to create [99, 264]. The difficulty depends on the data type and the features or materials that need to be classified. This issue is further discussed in Chapter 6.

LUTs offer a compromise by providing a certain pre-defined colour scheme that does not need to be configured by the user. The primary reason LUTs are useful for 2D slice visualisation is that occlusion is not a problem. Therefore, a colour gradient can be automatically assigned to the full dynamic range to enhance the contrast between different tissues or materials. The same is not possible during DVR, as both colour, and opacity must be configured.

As shown in Figure 5.25, LUTs can be used to visualise the attenuation gradient in a slice of an energy volume, or to display a concentration gradient in a slice of a material volume.

In MARS Vision, the LUTs are not hard-coded. Instead, they are stored in an XML (eXtensible Mark-up Language) file, which allows advanced users to create new tables and modify existing ones.

LUT design is nearly identical to the process of manually designing transfer functions: users identify (through a trial and error process) a colour scheme that shows a particular set of features well, and then save it for future use. A well-

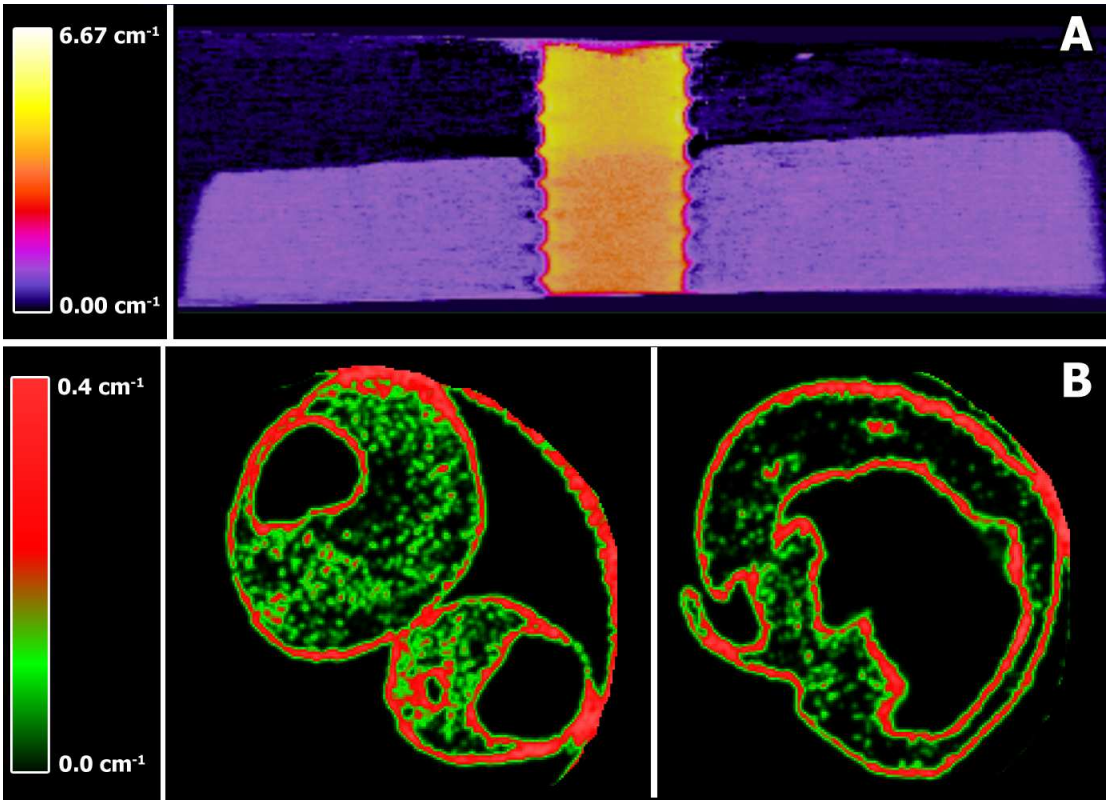


Figure 5.25: **A**: a slice of an energy volume of the TiScrew dataset (section 9.5.1) visualised using a "hot" LUT. **B**: two slices of the lipid channel of the Plaque77 dataset (section 9.2.1) visualised using a green-to-red LUT.

designed LUT is generic enough to be useful for visualising many different datasets. For example, a standard "hot" LUT (Figure 5.25, top), uses a range of colours in an attempt to maximise the visual difference between the high and low data values.

5.6.4 Spectral mode

The spectral mode is used to interactively fuse the slices of multiple channels of a spectral CT dataset. The need for 2D slice data fusion has been discussed in section 4.4.2. In brief, the fusion of two or more channels of a spectral CT dataset, such as overlaying material information onto a slice of an energy volume, has been often used to visualise spectral CT data (section 4.3). It is a useful technique, but there are no known custom GUIs or applications created specifically to facilitate spectral CT slice data fusion. Generic image processing software, such as ImageJ,

can be used, but the process involves manually setting the fusion parameters using a GUI that is not adapted for this task.

MARS Vision implements an algorithm that uses the transfer function editor described in section 6.4 to set the fusion parameters (colour and opacity). Figure 5.26 shows the primary steps carried out by the 2D spectral mode algorithm, while Algorithm 3 describes it in full.

To assign a colour to each pixel, the spectral mode algorithm samples all volumes, converts raw voxel values into RGB colour values using transfer functions, and blends all colours in the CIE XYZ colour space, using the opacity percentage for each channel as a weight. This process is a simplified version of MARS Vision's DVR algorithm.

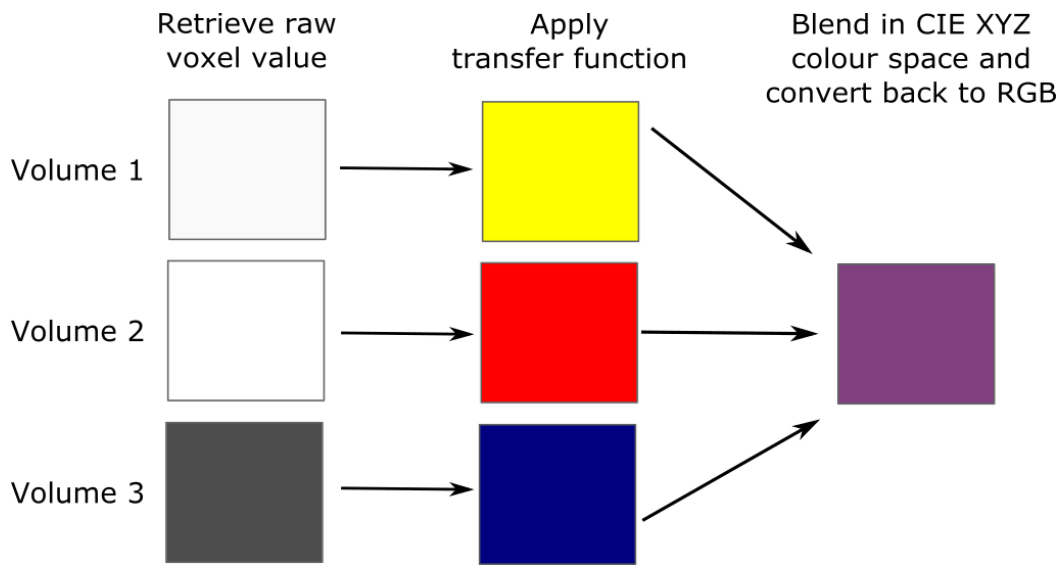


Figure 5.26: The spectral mode blending algorithm used to generate the colour value for each pixel of a slice.

The use of the same transfer functions synchronises the colour schemes used by 2D and 3D visualisation, as shown in Figures 5.27 and 5.28. The synchronisation is not perfect, as some difference in colour may be observed. The differences are caused by the DVR algorithm sampling a larger portion of the dataset and calculating directional and ambient lighting. These features are computationally intensive, and it is impractical to implement them outside of a DVR algorithm. In contrast, the 2D spectral mode algorithm only samples a single slice, but produces a similar result at a fraction of the rendering cost.

This is an important point, as all 2D visualisation algorithms in MARS Vision are executed on the CPU. This is done for compatibility reasons, as the DVR algorithm requires a compatible NVIDIA GPU, while the CPU code has no such restrictions. CPU-based 2D data fusion provides an alternative to the users who do not have access to powerful or compatible GPU hardware.

Algorithm 3 Algorithm for blending data for a single pixel during 2D spectral mode slice visualisation. All blending calculations are performed in the XYZ CIE 1931 colour space.

Data: voxel coordinates x, y, z; n volumes $V_{1...n}$; n transfer functions $T_{1...n}$
Result: pixel colour C

initialization:
$C \leftarrow$ black
temporary raw voxel value R
temporary colour TC
temporary opacity TO
total opacity $O \leftarrow 0$
loop through all volumes, retrieve the voxel values, apply transfer functions and blend:
for $i \leftarrow 1$ **to** n **do**
  $R \leftarrow$ retrieve voxel at coordinates x, y, z from V_i
  $TO \leftarrow$ determine opacity for R using T_i
  $TC \leftarrow$ determine colour for R using T_i
  $TC \leftarrow TC \cdot TO$ *multiply colour by opacity*
  $O \leftarrow O + TO$ *accumulate opacity*
  $C \leftarrow C + TC$ *accumulate colour*
end
divide by the accumulated opacity to produce a weighted average:
if $O \neq 0$ **then**
  $C \leftarrow C \div O$
end
save C

Spectral mode data fusion can be used to achieve a variety of effects. For example, it is possible to visualise all materials simultaneously, as shown in Figures 5.27 and 5.28. Alternatively, it is possible to overlay a material onto a single energy volume, (Figure 5.29) to show both the anatomical, and the material information. Similar approaches are common in dual-energy CT, PET-CT, and fMRI data visualisation, as discussed in Chapter 4.

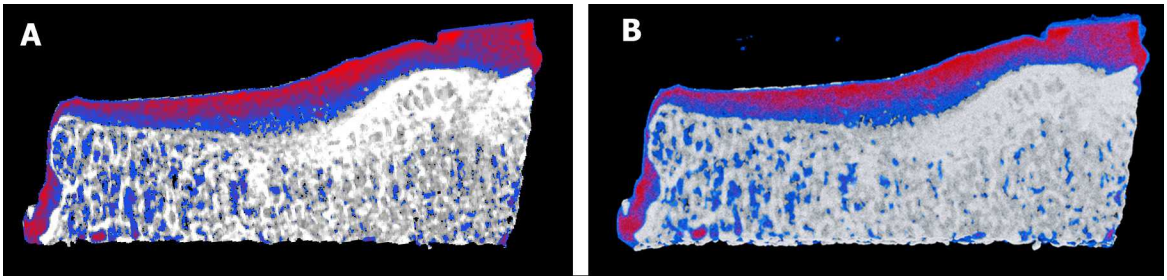


Figure 5.27: **A**: 2D visualisation of a slice of the Knee Cartilage dataset (section 9.4) in spectral mode. **B**: DVR of the Knee Cartilage dataset with the clipping planes set to show the same slice. The difference is due to the lack of shading in the 2D view.

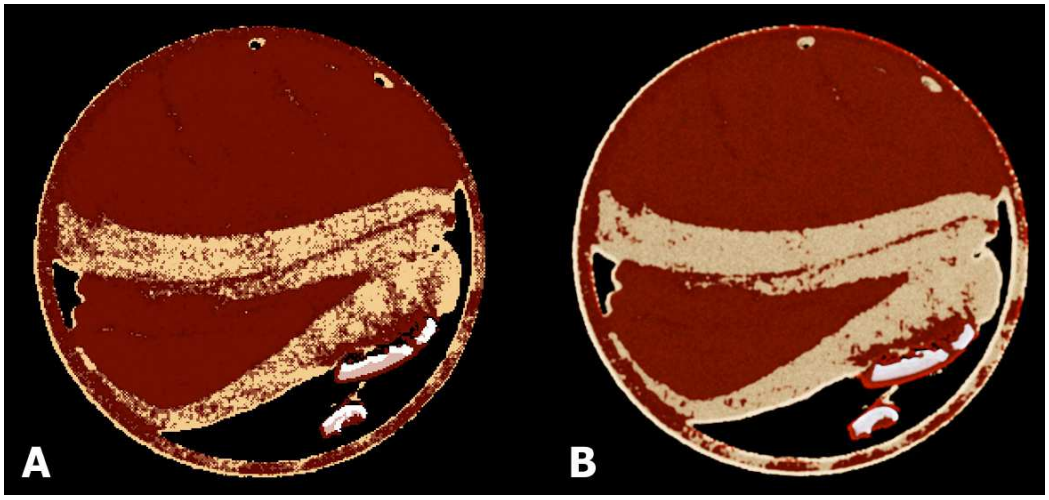


Figure 5.28: **A**: 2D visualisation of a slice of the Meat1127 dataset (section 9.4) in spectral mode. **B**: DVR of the Meat1127 dataset with the clipping planes set to show the same slice. Three materials are shown in both images (calcium - white, water - dark red, lipid - beige).

5.6.4.1 Colour replacement mode

The problem of the distortion of the colour gradient during DVR has already been addressed in section 5.3.5.1. The suggested solution is the colour replacement mode, which does not blend the colours calculated for multiple channels if the selected replacement channel is not transparent ($> 0\%$ opacity).

The colour replacement mode is also supported by the 2D spectral mode data fusion algorithm. The result is shown in Figure 5.30. Here, the colour assigned to barium channel (cyan/blue) is not blended with the colour assigned to the iodine channel (orange/red) if the opacity of the barium channel is greater than zero.

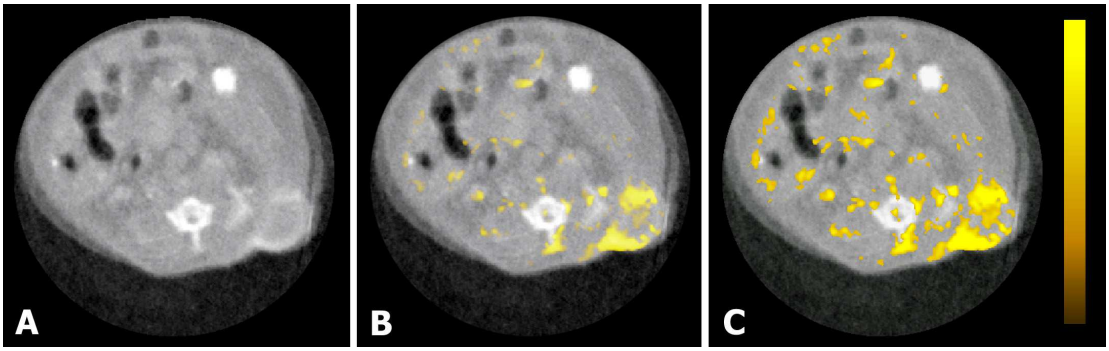


Figure 5.29: Spectral mode visualisation of the Mouse0 dataset (section 9.6.2). **A**: single energy volume, no data fusion. **B**: blending with the gold channel. **C**: colour replacement by the gold channel. The bar on the right shows the colour gradient assigned to the gold channel.

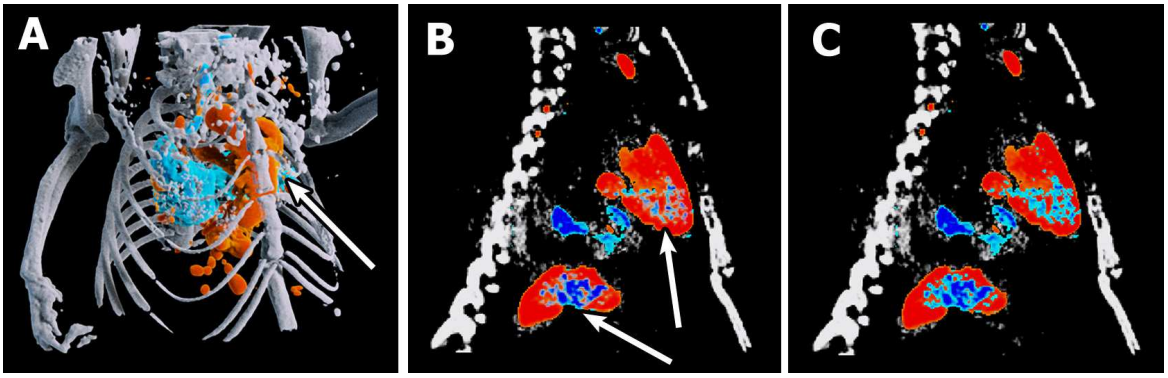


Figure 5.30: Visualisation of three materials from the Mouse12 dataset (section 9.6.1). **A**: 3D volume rendering with the ROI marked with an arrow. **B**: standard spectral mode blending. The areas where the iodine and barium channels overlap are marked with arrows. **C**: colour replacement by the barium channel.

Another example is shown in Figure 5.29C. This figure shows a common problem that occurs during the fusion of energy and material data. If an energy volume is visualised using a standard greyscale colour scheme (A), then blending will distort and darken the the colour gradient assigned to the material channel (B). Colour replacement avoids this problem (C).

5.6.5 Summary

This section has described the three 2D slice visualisation modes implemented in MARS Vision:

- Window and level. This mode uses the window and level parameters to display a slice of a single volume using the traditional greyscale colour scheme.

- Look-up table. This mode uses a pre-made transfer function to assign an arbitrary colour scheme to a slice of a single volume.

- Spectral mode. This mode uses the transfer functions (created using the editor described in the next chapter) to assign colour to each channel, fuses the data from multiple channels, and approximates the appearance of DVR.

5.7 Summary

This chapter has described the implementation of MARS Vision, a spectral CT data visualisation application that is based on the requirements described in Chapter 4. MARS Vision is integrated with the MARS software toolchain, provides a custom GUI for working with spectral CT datasets, and supports synchronised 2D and 3D visualisation of multiple energy and/or material volumes.

2D visualisation allows users to display slices of a single energy or material volume, or to fuse slices of multiple volumes, based on the colours assigned by transfer functions. 3D visualisation uses a novel Monte-Carlo ray tracing DVR algorithm capable of simultaneously rendering multiple volumes and correctly simulating the light transfer inside them. The algorithm maintains a depth buffer, which enables direct interaction with 3D visualisation, and is optimised to run on commodity GPU hardware.

This chapter has presented a novel methodology for fusing data from multiple energy and/or material volumes of spectral CT datasets. Data fusion is based on transfer functions, which are used to adjust the appearance of both 2D slice visualisation, and 3D DVR. This allows appearance of these two visualisation types to be synchronised, as the same transfer functions are used in both cases.

The algorithms described in this chapter solve an important research problem that has not been addressed until now. The fusion of spectral CT data during DVR has been studied before, but no research into extending traditional 2D tools to better visualise spectral CT datasets could be identified.

Another important issue addressed in this chapter is the distortion of the colour gradient during blending. The proposed solution is the colour replacement mode, which preserves the gradient assigned to one particular volume, while the data

from all other volumes is blended as usual. This chapter has demonstrated that the same colour replacement algorithm can be used for during 2D slice visualisation and 3D DVR.

The visualisation algorithms and software described in this chapter are used as a base for developing advanced tools for working with MARS spectral CT datasets. These tools are described in Chapters 7 and 8, while the next chapter discusses the implementation of a transfer function editor for working with spectral CT data.

Chapter VI

Transfer function design for spectral CT data visualisation

This chapter analyses the traditional approach to transfer function design and discusses a novel methodology for designing transfer functions for material volumes. It is structured as follows.

Section 6.1 explains the motivation for simplifying the process of transfer function design and discusses the advantages of creating transfer functions for material volumes. Next, section 6.2 describes the concept of a simple transfer function editor suitable for working with the material volumes produced by the MARS toolchain. Section 6.3 describes the integration of the simple editor with a traditional graph-based editor. Finally, section 6.4 describes the implementation of this tool in MARS Vision.

6.1 Background and motivation

The motivation for simplifying transfer function design be summarised as follows: in DVR, transfer functions are commonly used to classify volume data by assigning colour and opacity to different materials [60]. However, the process of designing transfer functions is complex, and the GUIs of transfer function editors require significant expertise to use [99, 264]. Clinical users are typically not trained in 3D visualisation or transfer function design, instead relying on 2D slice visualisation using a greyscale colour scheme controlled by window and level parameters [265]. Therefore, methods of simplifying the task of transfer function design should be examined. In particular, the new material data format generated by spectral CT MD offers an opportunity to simplify this process, as discussed in this chapter.

This section uses conventional CT datasets to illustrate the complexity of transfer function design. The same concerns apply to spectral CT visualisation, as each energy volume can be treated as a conventional CT volume (section 3.2.1). The design of transfer functions for material volumes is a different process, and will be discussed separately in the next section.

Section 2.2.2.1 has described the use of transfer functions to assign optical properties to certain data ranges during visualisation. These properties may or may not correspond to the physical properties of real materials. The mapping may be approximately realistic; alternatively, false colours may be used to emphasise certain features.

The common GUI for a traditional transfer function editor is based on a 2D graph that plots the data range along the x-axis and the opacity along the y-axis, as shown in Figure 6.1. The user is able to control both colour, and opacity using the same GUI element. Other types of transfer function editors decouple the adjustment of the colour gradient from the adjustment of the opacity gradient.

Regardless of the implementation, the basic principle is the same: the user is given the tools to choose a certain data range and assign a colour gradient, and an opacity gradient to it. This approach creates a one-dimensional transfer function for opacity and a one-dimensional transfer function for colour.

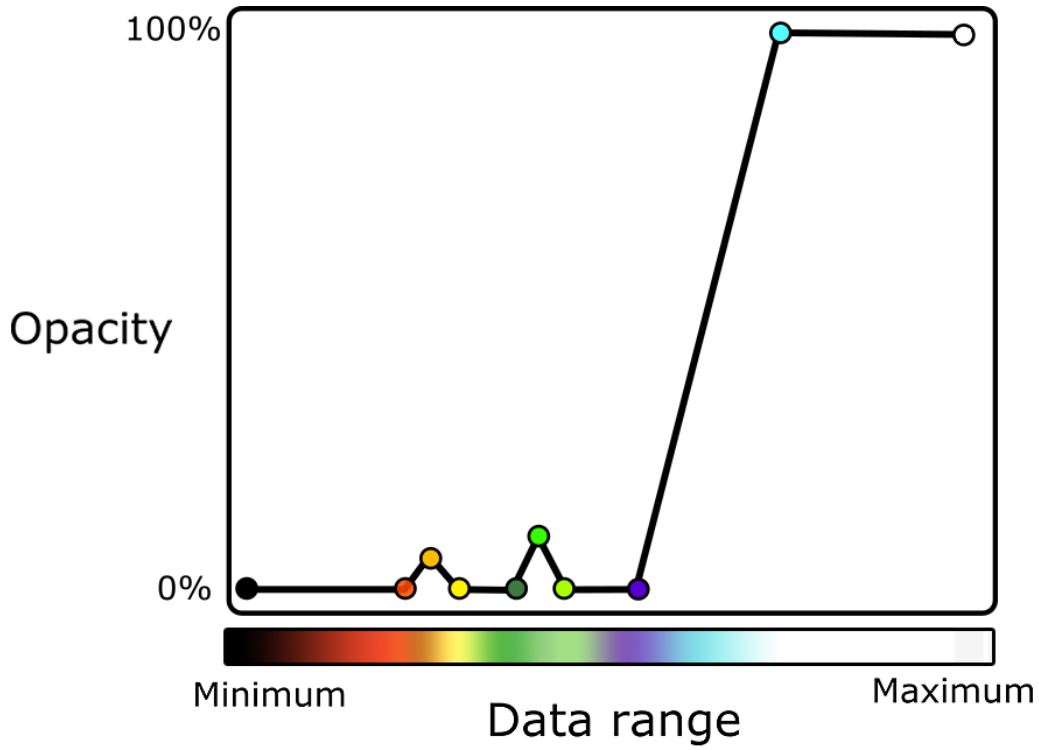


Figure 6.1: A traditional graph transfer function editor. Users must manually identify the data ranges that correspond to various tissues or organs and assign optical properties to these ranges by adjusting the position and colour of nodes on the graph. The colour gradient created by this graph is shown below it.

Multi-dimensional transfer functions are also possible [266,267,268], but largely restricted to computer graphics research. Medical visualisation is still reliant on 2D slice visualisation using the window and level parameters, and, since this approach is familiar to current MARS users, multi-dimensional transfer functions will not be considered at this time. However, such transfer functions have been successfully used for visualising other multi-modal datasets, so their application to spectral CT data should also be examined. This issue is further discussed in section 10.1.3.

The traditional graph editor displays the minimum data value on the left of the graph, and the maximum data value on the right, as shown in Figure 6.1. For example, the data range of conventional CT datasets that use the Hounsfield scale (section 3.2.1.1) will start at -1000 HU and end at around 3000 HU. Opacity is typically shown as a percentage.

The user places nodes on the graph and chooses a colour to be assigned to each point. The x coordinate determines the data value that is associated with that colour, and y coordinate determines the opacity. Colour and opacity are linearly interpolated between the nodes. For example, the colour for the data values that lie between the eighth and ninth points on the graph in Figure 6.1 will be interpolated between dark purple and cyan, while the opacity will be interpolated between 0 and 100%.

Therefore, a standard transfer function used in DVR is simply a mapping from a certain data value (for example, a linear attenuation coefficient, Hounsfield Unit, or concentration) to a colour (usually in the RGB colour space) and an opacity percentage. This mapping is shown under the graph in Figure 6.1, where the colour gradient created by this transfer function is shown as a colour bar. This means that values close to the minimum of the data range will be mapped to black, values close to the maximum will be mapped to white, and the various data values between the minimum and the maximum will be mapped to colours such as red, orange, green, and so on.

Graph transfer function editors are present in several freely-available open-source volume rendering applications (for example, MITK [69], 3D Slicer [68] and Voreen [107]). Figure 6.2 shows the graph transfer function editors implemented in MITK and Voreen. However, such editors are not commonly used in clinical practice because transfer function design is often a complex, manual, and time-consuming task [100].

In theory, the user may attempt to visually separate tissues or materials by

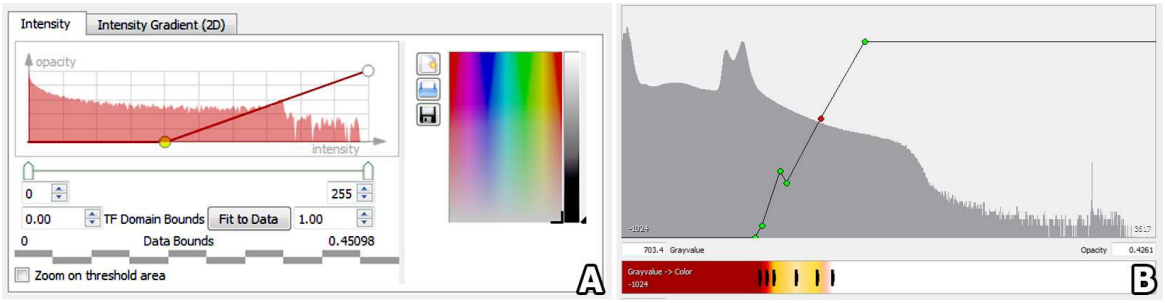


Figure 6.2: **A**: transfer function editor implemented in MITK. **B**: transfer function editor implemented in Voreen.

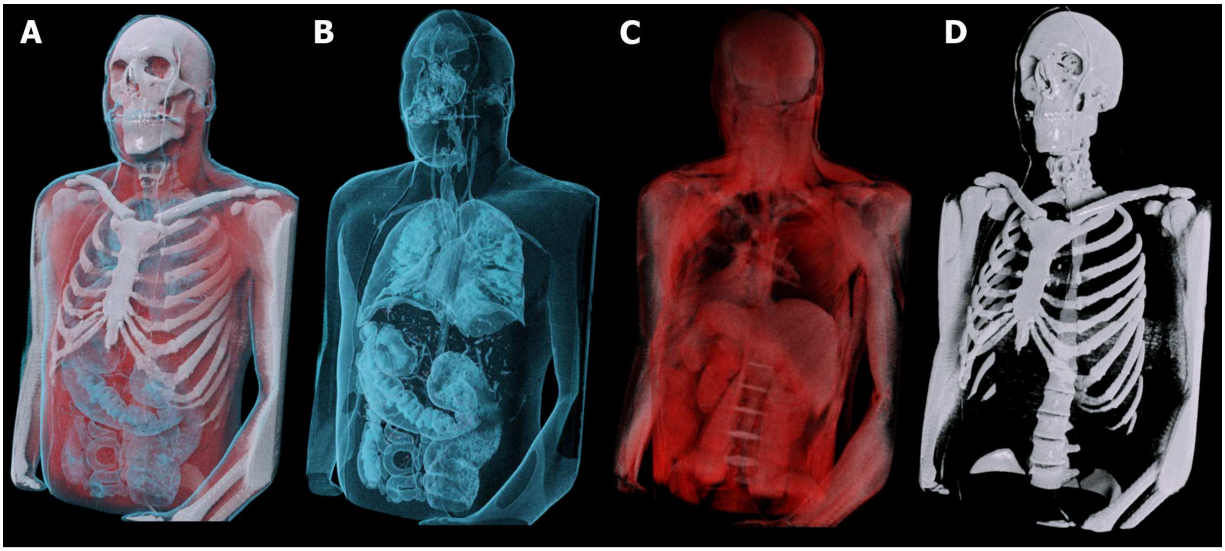
identifying the correct attenuation ranges and assigning unique colours, as shown in Figure 6.3. However, in practice, this may not always be possible, as the attenuation ranges for different tissues may be very similar, or even overlap. In addition, the user must manually adjust the position and colour of each node on the graph until the desired appearance is achieved.

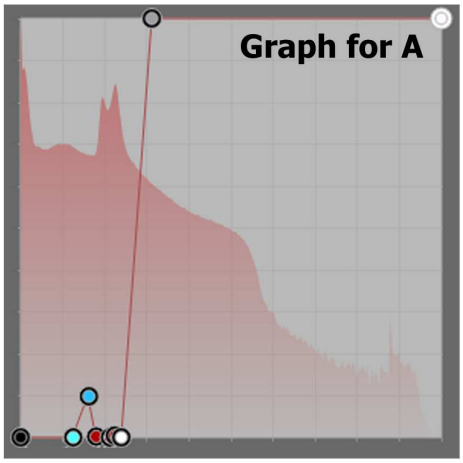
Identifying tissues in clinical CT datasets by analysing the differences in attenuation can be a complicated task. The reason is that multiple tissues may have the same HU value. For example, the typical HU values for cerebrospinal fluid (CSF) range between 5 and 20 HU, while soft tissues range between 20 and 50 HU, blood between 50 and 100 HU, and white and grey matter between 25 and 40 HU [269]. Common image artefacts, such as beam hardening, can lead to further distortion of these values.

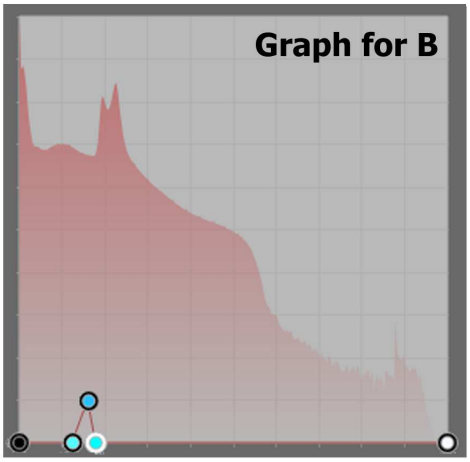
Therefore, accurate manual identification of a tissue from its HU value may not be possible. As stated in section 3.2.1.1, the MARS team does not use the Hounsfield scale to store energy volume data; instead, linear attenuation coefficients are used. However, the same problem remains, with multiple tissues potentially sharing the same linear attenuation range.

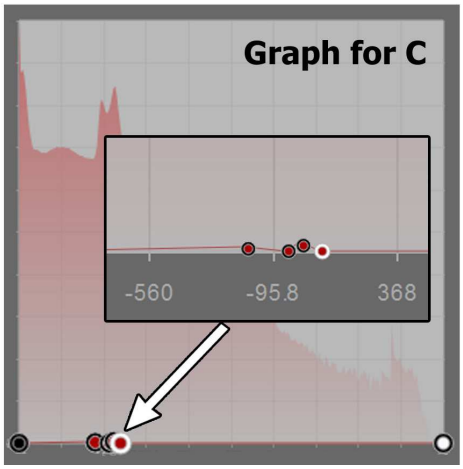
Figure 6.3 can be used as another example of the difficulty of manual transfer function design. Figure 6.3C shows the soft tissues in a full-body CT dataset as a translucent cloud. Consider the transfer function necessary to achieve this visual effect: a small data range (magnified in the inset) must be manually identified as representing the soft tissues in this dataset. In addition, the opacity for this data range must be set very low, otherwise the soft tissues will occlude the bones, if they are visualised concurrently, as is done in Figure 6.3A.

It must be noted that transfer function design can be simplified through the









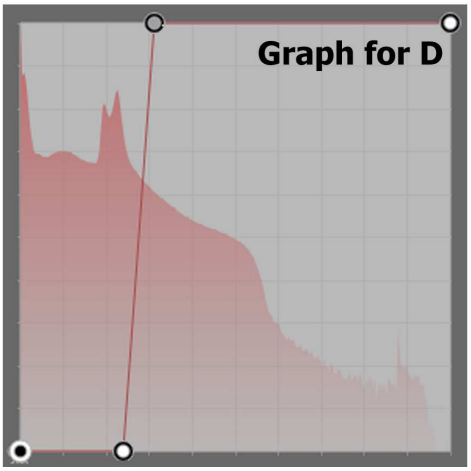


Figure 6.3: Visualisation of the Visible Human Male dataset (section 9.7.1). **A**: transfer function classifying the boundary between air and soft tissues (cyan), soft tissues (red), and bones (white). **B**: the boundary between air soft tissues only. **C**: soft tissues only. **D**: bones only.

use of presets, which are pre-defined settings that emphasise certain features, while hiding others. First, a visualisation expert may design a preset that shows a particular set of structures. Inexperienced users may then use this preset as a base and adjust it to fit their needs.

In this case, established medical imaging modalities, such as single-energy CT or MRI, have an advantage over spectral CT, as the data formats have been standardised. A good example of this is the near-universal use of the Hounsfield scale in clinical CT imaging. This standardisation means that the ranges of Hounsfield Units that correspond to certain organs or tissues are well-known (though, of course, it does not solve the problem of different tissues having similar HU values, as mentioned above). Presets that show these tissues can be created and applied to different datasets [270, 271]. This cannot yet be done for spectral CT, as there is currently no standardisation of data formats.

6.1.1 Properties of material volumes that enable a different approach to transfer function design

Transfer functions are used to classify tissues or materials present in conventional CT datasets, or energy volumes of a spectral CT dataset. However, it is a purely visual classification that provides no quantitative information about the concentrations of materials. In addition, it relies on the user correctly identifying the attenuation ranges that correspond to various tissues or materials. If a range is identified incorrectly, then the apparent amount and/or distribution of a material will not correspond to the real amount.

In contrast, MARS MD algorithms both classify, and quantify materials, as explained in section 3.2.2. These properties can be taken advantage of to simplify the process of visualisation and transfer function design.

When working with energy volumes, users must manually identify materials. Therefore, the resulting transfer function and the accuracy of material identification depends entirely on the user's skill. In contrast, MD automatically identifies materials and places them into separate volumetric datasets. The accuracy of MD depends on the algorithm and the quality of source energy volume data, but never on the user's skill.

Materials are stored as separate volumetric datasets, where each voxel represents the concentration at a certain position. Therefore, all variation in the voxel

values of a material volume is solely due to the differences in concentration. An example is shown in Figure 6.4. The colour gradient shown in these slices directly corresponds to the concentration gradient in the material volumes.

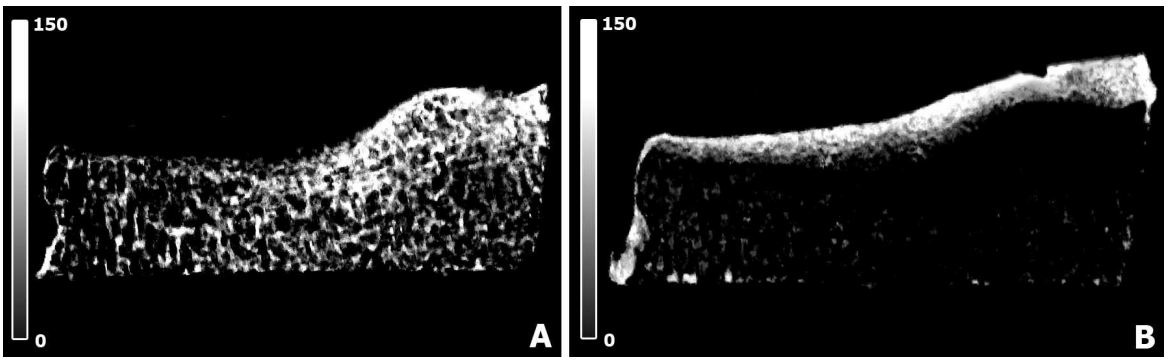


Figure 6.4: The calcium (**A**) and iodine (**B**) extracted from a scan of the Knee Cartilage dataset (section 9.4). The concentrations are given in mg/ml.

In contrast, an energy volume stores the attenuation of all materials present in the scanned object. Different materials may have similar attenuation profiles. The measured attenuation coefficient of a voxel may be caused by the presence of several materials. This complicates material separation and requires the use of complex GUI elements, such as the graph transfer function editor.

Therefore, traditional transfer function design for CT datasets faces two problems: classification of materials, and creating gradients to visualise the changes in their density. The user must set the boundaries of the data ranges that correspond to each material, and then assign a colour and opacity gradient over each of these ranges. Solving this problem without further data processing (such as automated segmentation of the dataset [176,272,273,274]) requires an advanced GUI. In comparison, when working with material volumes, the concentration of a material is the only variable.

In conclusion, from the user's point of view, MD simplifies the representation of the scanned object, as the materials comprising it are separated and quantified. This allows for the development of a custom transfer function editor for working with material volumes.

6.1.2 Summary

This section has discussed the difficulty of transfer function design for traditional CT energy volumes and explained the reasons why traditional transfer function editors are unnecessary for material volumes. Instead, a simplified approach that directly maps a concentration gradient to a colour gradient can be used. The algorithm and GUI for performing this mapping are described in the next section.

6.2 Simplified transfer function editor for material volumes

This section describes a tool, referred to as the simple transfer function editor, for creating transfer functions for material volumes of a spectral CT dataset. It has been developed in response to the feedback gathered from MARS users, who were not familiar with the process of transfer function design using a traditional graph-based GUI (as mentioned in section 4.3.2.1).

This editor is the first identified tool that has been developed specifically for creating transfer functions for spectral CT material volumes. The basic concept behind it is the application of the commonly-used concept of window and level settings, described in section 2.2.1, to material data visualisation.

Window and level adjustment is a standard tool in medical imaging [20], and a near-universal feature of DICOM viewers [81, 238], and scientific visualisation software such as ImageJ [67]. In addition, almost all identified studies into any aspect of spectral CT (detector design, denoising algorithms, reconstruction, material decomposition, or clinical applications) have used 2D slice visualisation and adjusted the appearance of slices using window and level settings [5, 7, 9, 12, 13, 48, 55, 57, 58, 220, 221]. This includes the work published by the MARS team.

Therefore, we can conclude that the vast majority of researchers currently working with spectral CT are familiar with the idea of creating a greyscale colour scheme using two adjustable parameters. The transfer function editor presented in this section is specifically designed to exploit this prior knowledge possessed by the users.

The concept of a simple transfer function editor is shown in Figure 6.5. This GUI acts as a proxy for creating a traditional graph transfer function. The complexity of a traditional GUI, however, is not exposed to user, who adjusts the appearance of a volume through a set of sliders and colour selection dialogs. This

GUI allows the user to assign a colour gradient to a particular data range and to control the visibility of each material channel.

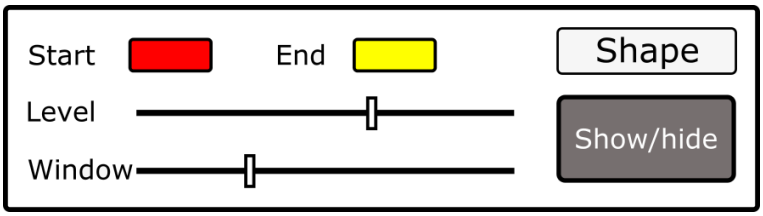


Figure 6.5: The initial concept of a transfer function editor for material volumes. The appearance of each volume is controlled by the level and window sliders and by the start and end colours. A colour gradient is created based on the window and level settings.

The *level* setting controls the position of centre of the window and the *window* setting controls its width. The *start* colour is assigned to the data value at the beginning of the window, and the *end* colour is assigned to the data value at its end.

Figure 6.6 shows that these settings can be mapped to different shapes of the transfer function graph. Both the colour, and the opacity gradients can be varied. This figure provides three examples, though other shapes are also possible. The shape is changed using the shape selector shown in Figure 6.5. In practice, this selector can be implemented as a GUI widget such as a combo box.

The first and most basic type of transfer function is the ramp (Figure 6.6A), where the opacity is 0% before the start of the window. The opacity is then interpolated between 0 and 100% over the width of the window, and remains at 100% until the end of the data range. The level controls the position of middle of the slope, and the window controls the steepness of the slope. This shape creates a colour and an opacity gradient.

The second type is the solid colour gradient shape (Figure 6.6B), which is the same as the ramp shape, but provides no opacity gradient. This is useful when a concentration gradient needs to be shown, but the opacity of a material must remain constant.

The final example is the window shape (Figure 6.6C), where the opacity of the data is 0% everywhere, except inside the window. The level parameter controls the position of the cea lowentre of the window, and the window parameter controls its width. A colour gradient spans from the beginning to the end of the range, but an opacity gradient is absent.

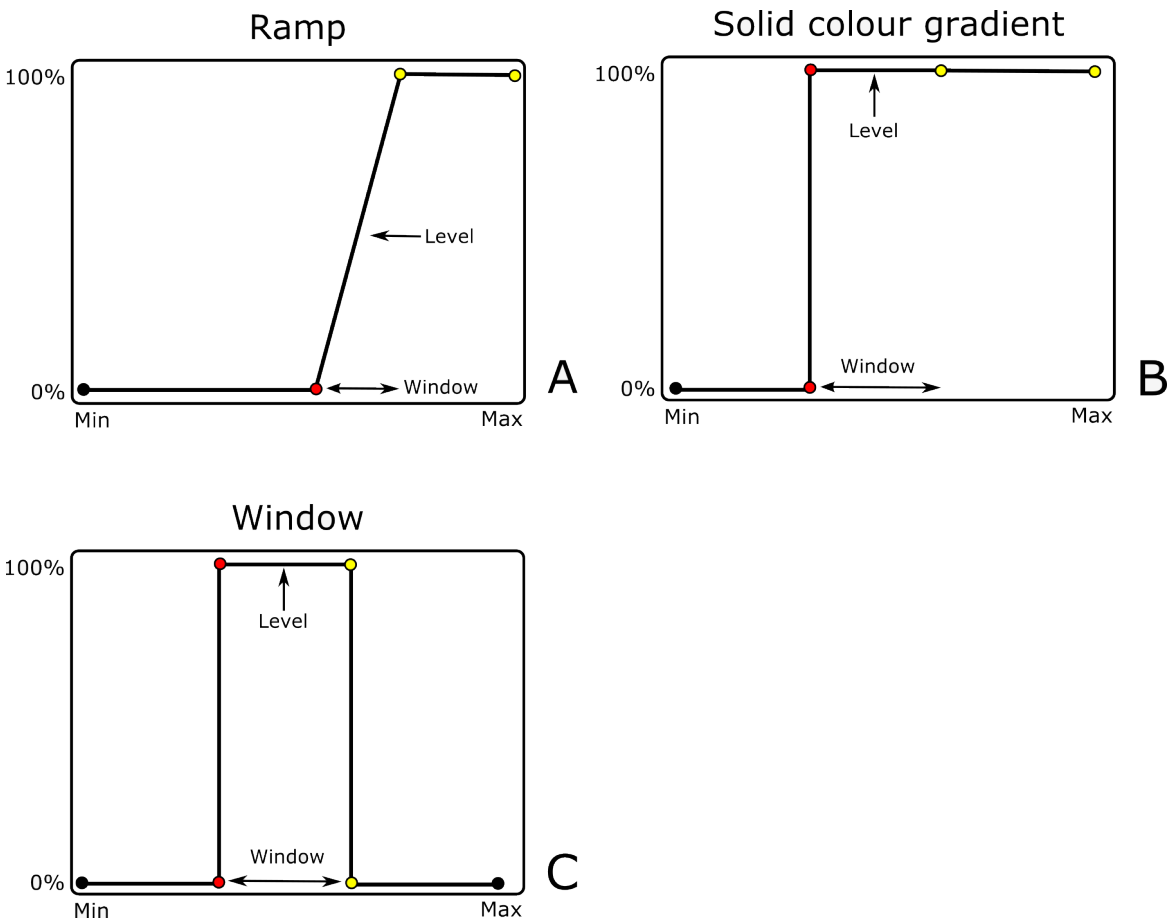


Figure 6.6: The mapping of window and level parameters to different transfer function shapes. **A**: ramp. **B**: solid colour gradient. **C**: window.

The ramp and solid gradient shapes cannot be used to show low concentrations of a material, while hiding higher concentrations. This is because the opacity gradient is pre-determined: it always begins at 0% at r concentration, and always ends at 100% at a higher concentration. This is can be compared to the mapping created by window and level settings during 2D slice visualisation: lower data values are always shown using a darker colour than higher values. In contrast, the window shape allows a certain concentration range to be displayed, while all other concentrations are hidden, as shown in Figure 6.7.

6.2.1 Extended version

This GUI described above can be used to adjust the appearance of a single volume. However, spectral CT data visualisation may involve simultaneously displaying

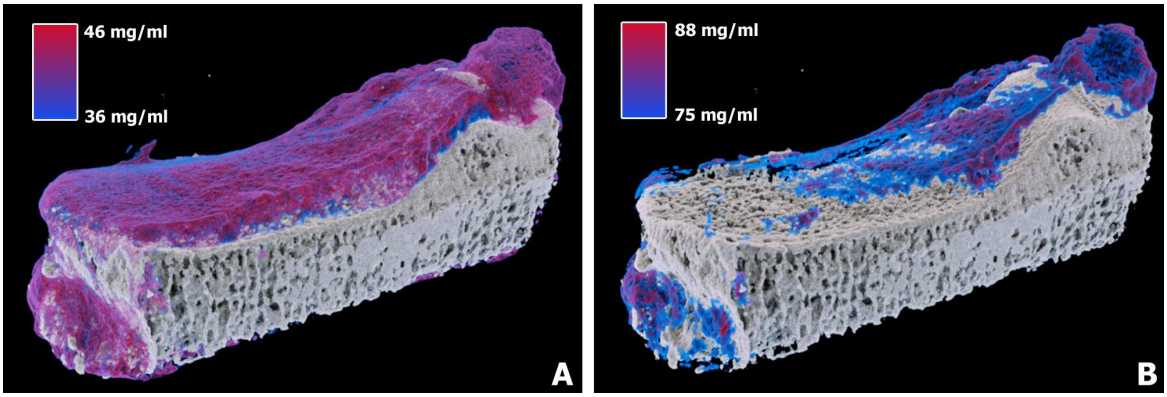


Figure 6.7: Displaying two different concentration ranges of the iodine channel of the Knee Cartilage dataset (section 9.4) using the window transfer function shape.

multiple volumes. Therefore, it is not enough to assign colour and opacity to a single volume, and we must also consider the interaction between the volumes.

The editor shown in Figure 6.5 allows the user to hide or show a material channel by using a dedicated button. However, this GUI is insufficient when multiple volumes are overlapping, or occluding each another, as it does not allow for the precise control of a material's transparency.

Therefore, a third parameter needs to be added. In practice, it can also be controlled with a slider. The third parameter controls the opacity of a material channel and removes the need for a button to toggle its visibility. Now, a volume is considered to be completely invisible when its transparency is 0%. The extended version of the simple transfer function editor for material volumes is shown in Figure 6.8.

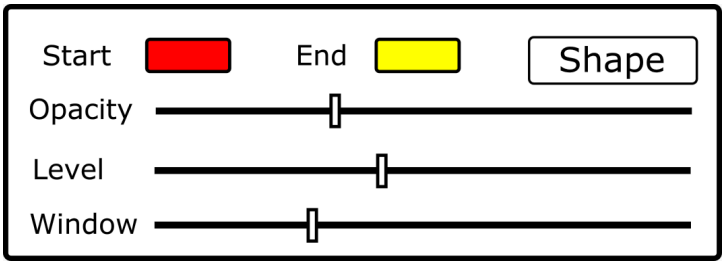


Figure 6.8: The final concept a transfer function editor for material volumes. The GUI is nearly identical to that in Figure 6.5, with the exception of an additional slider to control the opacity of the volume. The button used to show or hide the volume is no longer required.

The final issue to consider is the visual feedback provided to the user. The implementation proposed in Figure 6.8 does not display the colour gradient created by the window and level parameters, so the user may struggle to understand how changes to these parameters affect the generated colour scheme. This problem can be addressed by displaying a colour bar that represents the gradient. The bar may be located in the transfer function editor, as shown in Figure 6.9. Alternatively, it may overlay the visualisation canvas, as shown in Figures 5.25 and 5.29.

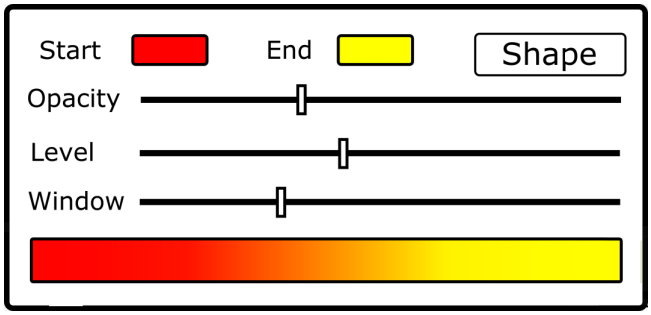


Figure 6.9: One possible way of displaying the colour gradient created using the simple transfer function editor.

Section 6.4 describes the implementation of this GUI concept in a spectral CT data visualisation application. Chapter 9 provides examples of the use of this editor for visualising MARS spectral CT datasets. It discusses the uses of the different transfer function shapes and demonstrates that this GUI is sufficient for designing transfer functions in some current application scenarios.

6.2.2 Discussion

The simple transfer function editor is the first identified tool for adjusting transfer functions for material volumes of a spectral CT dataset. This is not surprising, as the majority of the studies into the clinical applications of spectral CT (reviewed in section 4.3) used 2D slice visualisation. Appearance was adjusted using a greyscale colour scheme controlled by window and level settings. No custom GUI elements were needed to assign colour, as the fusion of data from multiple channels was not commonly performed.

The MARS team has a slightly different approach. Like the other research teams, it has often visualised slices of energy and material volumes separately, but it has also studied the methods for fusing and simultaneously displaying the

information from multiple channels. This thesis has done so too, presenting the algorithms for DVR and 2D fusion of spectral CT data in Chapter 5. However, the interface for controlling the colour scheme, visibility, and other parameters, is also vitally important, as it is the only way for the user to interact with the underlying rendering algorithms. The design of such interfaces was not previously a priority for the MARS team, which was the motivation for developing this GUI during my PhD research.

As mentioned before, this user interface has been created in response to the feedback from current MARS users, who were not familiar with the GUI and data intermixing techniques implemented in MARSCTExplorer (section 4.3.2.1), the first visualisation application developed by the MARS team. It attempts to use the well-known concept of window and level adjustment for setting colour and opacity parameters that can be used for both 2D, and 3D visualisation of material volumes. Further, this editor is a very compact GUI element: several editors can be displayed on the screen to allow the user to quickly view the colour gradients assigned to every material volume in the dataset. See sections 6.3, 6.4, and Chapter 9 for examples.

The proposed transfer function editor is, of course, limited compared to the traditional graph-based interface. However, these two editors are designed for working with completely different data types:

- The graph editor is required for working with CT energy volumes (as well as with other common data types, such as MRI datasets) because it is a good tool for manually extracting features from a dataset that contains multiple materials or tissue types. This task is absolutely essential for visualising CT energy volumes. Therefore, an editor that allows the user to precisely designate multiple data ranges and assign different optical properties to each one is needed.

- The simple transfer function editor is specifically designed for material volumes, which have already been processed by material decomposition algorithms. Its capabilities are heavily restricted because, in the currently-known usage scenarios, assigning a colour and an opacity gradient to a certain concentration range is all that is required. Nevertheless, it should not be the only transfer function editor available, as advanced users may desire a greater degree of control over the visualisation parameters. This is the reason for

fusing the simple and the graph editors into a "hybrid" transfer function editor, described in section 6.3.

The simple transfer function editor is most similar to the high-level semantic interface proposed by Rezk-Salama et al. [275], which allowed the user to control the appearance of several tissue types (for example, skin, bone, blood vessels, and soft tissues) using colour selection dialogs and sliders. Each tissue type was controlled using a single colour, a visibility slider, and an "adapt template" slider, which was used to adjust the criteria used for automatically identifying this tissue type. The transfer function used for DVR was created based on these settings, but was never shown to the user.

This concept is simple, but, at this time, insufficient for spectral CT data visualisation. The reason is that it is too high-level, and only treats the dataset as a collection of tissue types. There are only two options: a voxel either belongs to a particular tissue, or it does not. However, in MARS datasets, voxels may belong to several material volumes at once (which means that there are multiple materials inside this voxel). Therefore, when visualising MARS material volumes, the user needs to make more detailed choices, which include selecting the materials to be shown, settings the colours or colour gradients, configuring the opacity, and then controlling the visible concentration range for each material. The proposed transfer function editor based on window and level parameters offers this flexibility, while still retaining the user-friendliness of slider-based GUIs.

Finally, it must be noted that the concept presented in this section is largely independent of the type of technique used for displaying the data. As shown in Chapter 5, the same transfer functions can be used to assign colour and opacity during both 2D slice visualisation, and 3D DVR. The algorithms must fulfil some criteria, such as support for multiple volumes (for both 2D, and 3D techniques) and per-volume transparency (3D techniques only), but, as discussed in section 4.4, these are very basic features that most algorithms for spectral CT data visualisation are expected to possess.

6.2.3 Summary

The simple transfer function editor is a GUI for mapping a colour gradient and/or an opacity gradient to a range of material concentrations. It applies the concept

of window and level adjustment to material data visualisation, as these settings are used to create a colour and an opacity gradient over the chosen data range.

Interaction with the simple transfer function editor is conducted through two colour selection dialogs and three sliders (for adjusting window, level, and the overall opacity of the volume). The underlying algorithm creates a traditional transfer function graph, but this information is not presented to the user, who is not expected to be familiar with transfer function design. The user's choice of transfer function settings is intentionally limited, since the arbitrary mapping of colour and opacity with the graph editor is not a well-known technique in medical imaging.

6.3 Hybrid transfer function editor

The transfer function editor described in the previous section is intended for working with material datasets only, as the restrictions on the shape of the transfer function and the complexity of the colour gradient make it ill-suited for working with energy volumes. However, current MARS datasets combine energy and material channels, as it is useful to use an energy channel (which contains all data) as a basis to map the material on. In other words, the energy channel is used as the context. This is also a common technique in PET-CT and dual-energy CT data visualisation, as discussed in sections 4.1 and 4.2.

Therefore, the user must be given the tools to design transfer functions for both data types. Inexperienced users may choose to only use the simple editor. Advanced users may wish to use the graph editor to create complicated transfer functions and visual effects. The hybrid GUI proposed in this section accommodates both workflows.

The two transfer function editors can be combined into a single GUI element as shown in Figure 6.10. This layout is only a suggestion, and the exact layout can be adapted to fit the particular application.

This combination of two transfer function editor types is possible because both generate one-dimensional transfer functions for colour and opacity. The simple editor, as shown in Figure 6.6, can produce a range of fixed different graph shapes, but the same graphs can also be manually created by using the traditional graph GUI. The reverse mapping is not always possible because the graph editor can create an arbitrarily-complex transfer function that cannot be mapped to the level

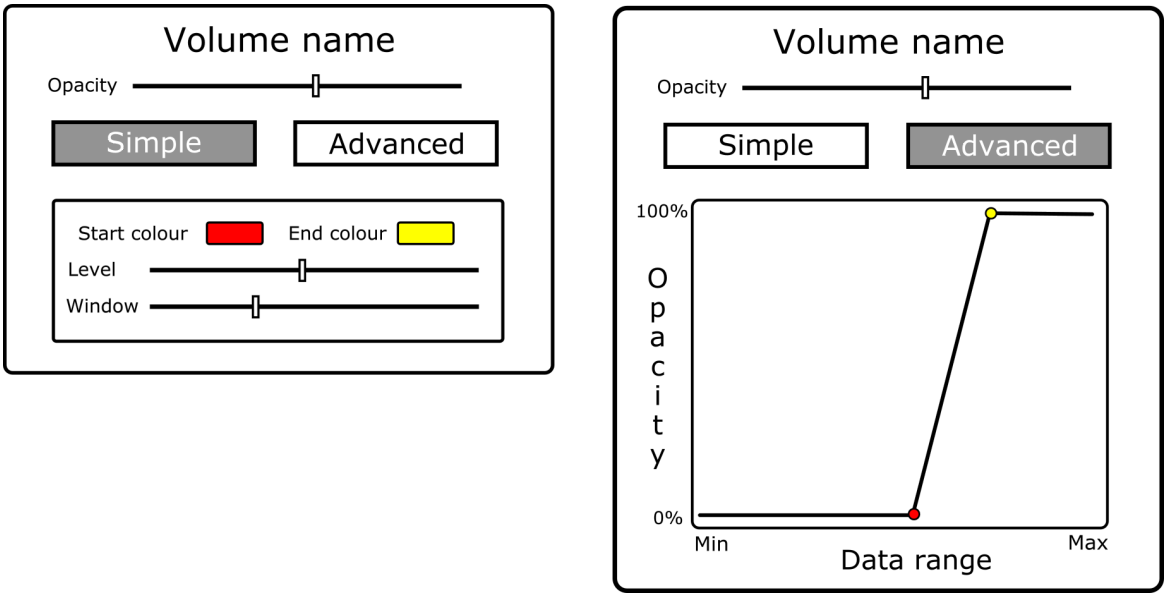


Figure 6.10: The concept of the hybrid transfer function editor GUI. Left: The simple form. Right: the advanced form. The user is able to toggle between the two forms. Note that the opacity slider is present in both forms.

and window parameters.

The hybrid editor contains all GUI elements needed to create a transfer function for an energy or a material volume, and to adjust its opacity. Therefore, the last issue to consider is the placement and layout of multiple hybrid editor widgets in the GUI of a spectral CT data visualisation application.

Figure 6.11 proposes two possible layouts. Placing the editors in a list allows the user to view the transfer functions for several volumes at once. However, this arrangement consumes a significant amount of vertical space. Scrolling through the list may be required if the dataset contains a large number of volumes. On the other hand, placing the editors in a tabbed interface is much more compact, but only shows a single editor GUI at a time.

6.3.1 Summary

The hybrid transfer function editor is a combination of the simple transfer function editor proposed in section 6.2 and the traditional graph-based editor GUI. Each volume of a spectral CT dataset is associated with a separate hybrid editor, which maintains a transfer function for that volume. Users can choose the most appropriate form based on their skill level, the requirements of the particular visualisation problem, and the volume's data type.

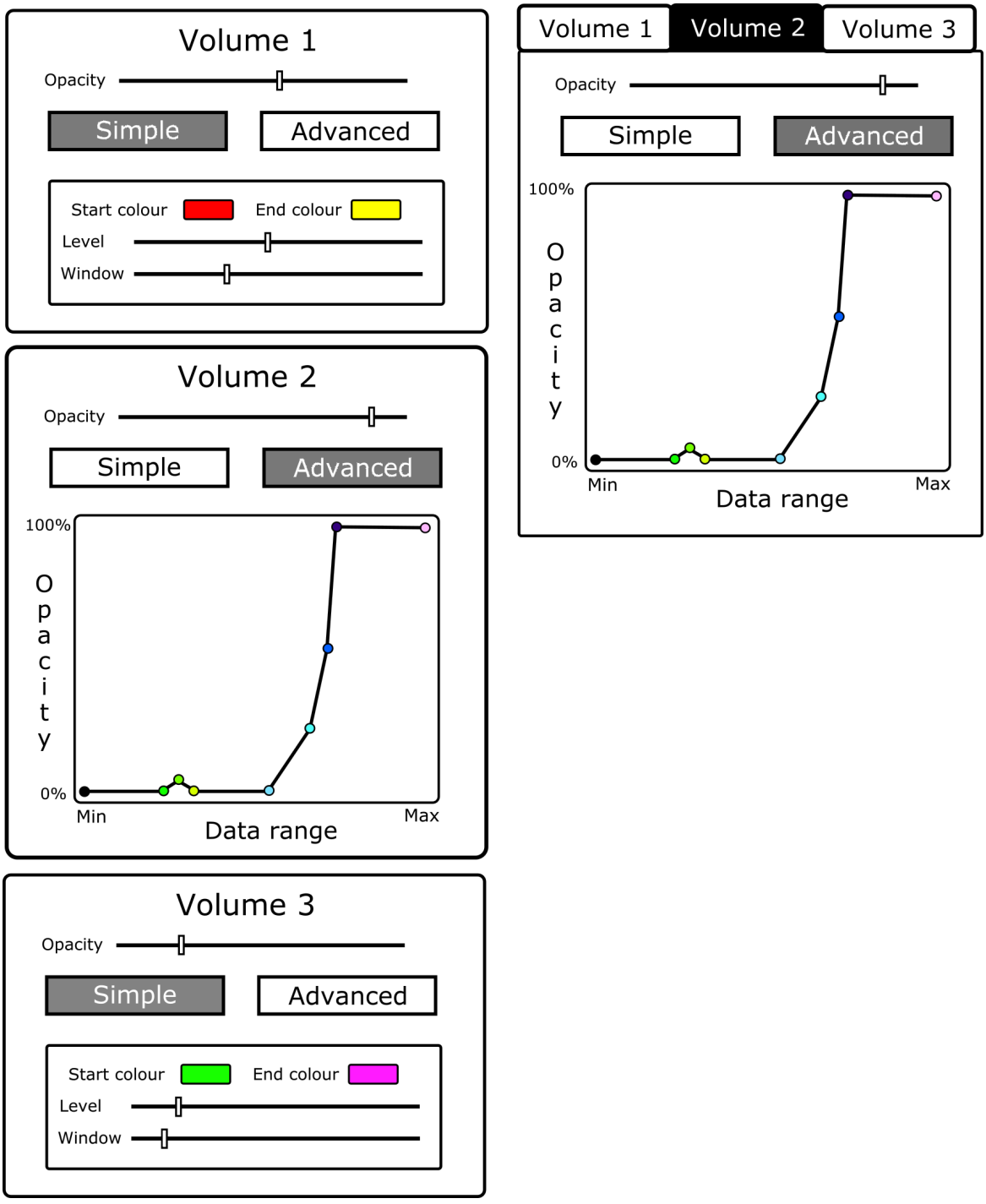


Figure 6.11: Two possible arrangements of hybrid transfer function editors. Left: vertical list. Right: tabbed interface.

6.4 Implementation in MARS Vision

Previous sections have described the concept of a hybrid GUI for designing transfer functions for both energy, and material volumes of a spectral CT dataset. The GUI can be toggled between two forms, allowing the user to choose the appropriate form for each volume. This section describes the implementation of this transfer function editor in MARS Vision.

In MARS Vision, each transfer function editor is linked to a set of buffers on the GPU. The buffers store the volume data array along with the transfer function to be applied to it during rendering. Each transfer function editor also maintains a separate opacity parameter that can be used to adjust the opacity of the entire volume.

The implementation of the transfer function editor GUI is shown in Figure 6.12. For practical reasons, it modifies the concept presented earlier in this chapter. In particular, the volume associated with each transfer function can be changed, as shown in Figure 6.13. This is done because:

- It allows greater flexibility during the intermixing of volume data. For example, the same volume can be rendered twice, using two different transfer functions that show different features.

- The amount of GPU VRAM is limited. In comparison, the amount of system RAM is usually several times larger. Therefore, it may be possible to hold a large number of volumes in RAM, but it may not always be possible to render all volumes simultaneously.

  This situation is illustrated in Figure 6.13, where the dataset comprises four volumes, but the GPU is only able to store three. By default, MARS Vision will allocate three GPU buffers and load the first three volumes into them (A). However, the user may then change the volume that is loaded into each buffer (B, C).

The second change concerns the advanced form of the transfer function editor (Figure 6.12C), which retains the window and level sliders used by the simple form. This means that the user can adjust the sliders and observe the effects by looking at the changes of the transfer function graph.

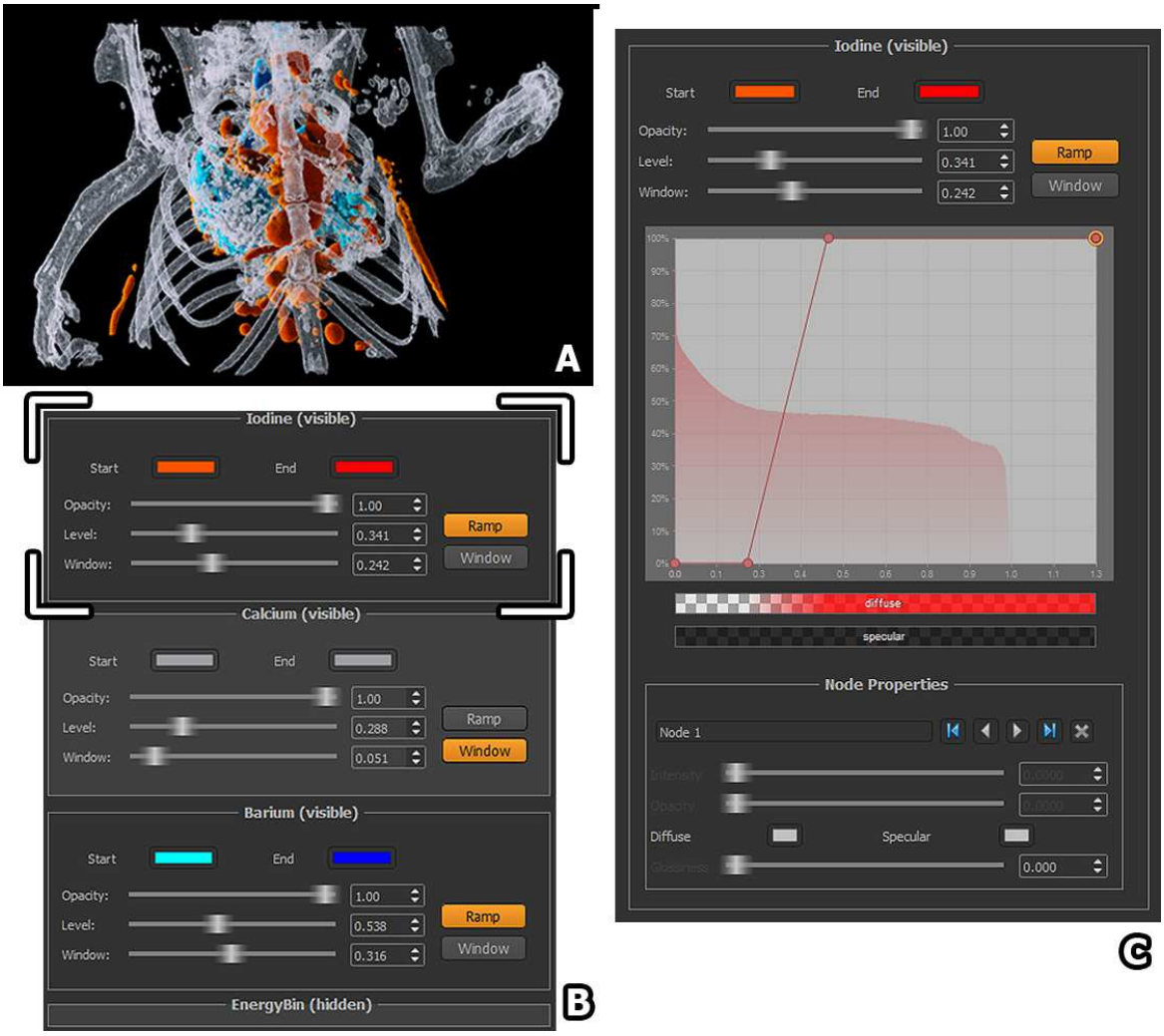


Figure 6.12: Implementation of the hybrid transfer function editor in MARS Vision. **A**: the result of visualising the Mouse12 dataset (section 9.6.1) using the parameters shown in **B**. **B**: the simple form of the hybrid transfer function editor. The four editor GUI widgets are laid out in a vertical list. Note the buttons used for switching between the ramp and the window transfer function shapes. **C**: the advanced form. The GUI for editing the transfer function graph and the properties of a node has been implemented by Kroes et al. in the original Exposure Render.

6.4.1 Mapping from the advanced to the simple editor

Section 6.3 has argued that it is not always possible to map a transfer function created with a traditional graph editor to the window and level parameters used by the simple editor. Usually, this is true, as the simple editor is only capable of creating fixed-form transfer function shapes, while the graph editor has no such

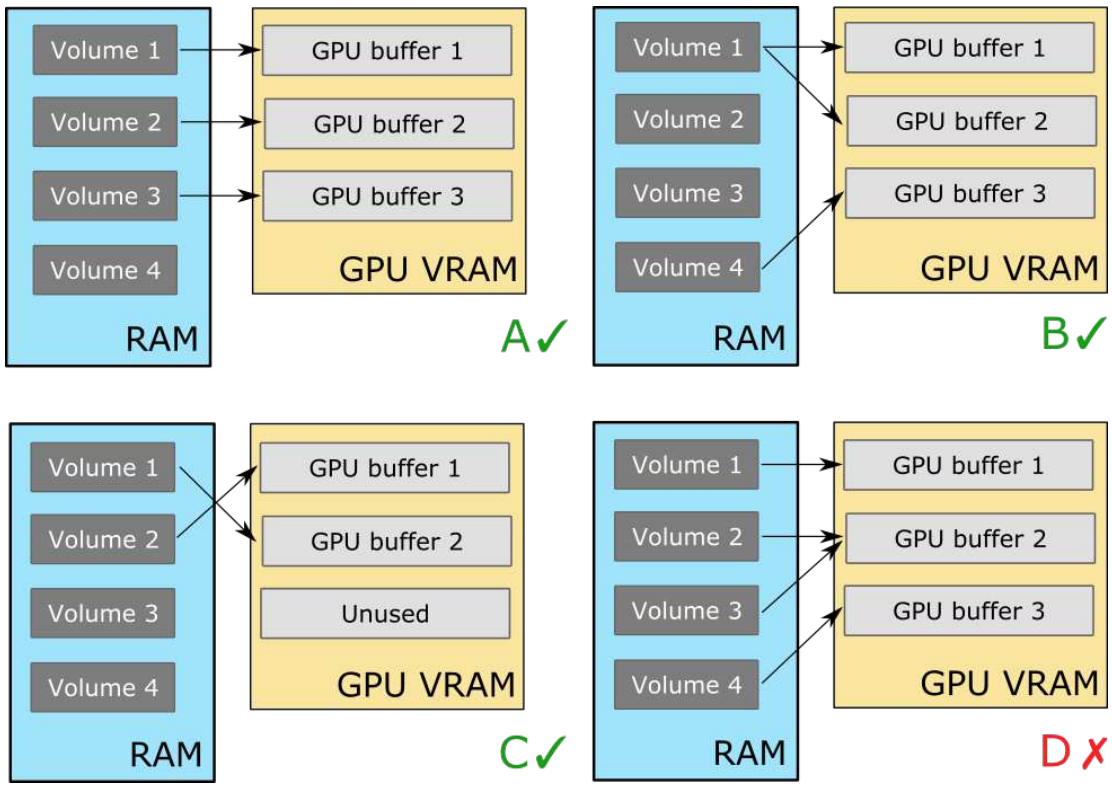


Figure 6.13: The mapping of volumes to GPU buffers in MARS Vision. Each buffer is associated with a transfer function which is applied to the volume stored in that buffer. **A** shows the default configuration. However, a volume may be loaded into multiple buffers simultaneously (**B**), or assigned to a different GPU buffer (**C**). Buffers may also be left empty (**C**). However, a buffer may only hold a single volume (**D**).

restrictions.

However, in some cases, the shape of the transfer function graph can be mapped back to window and level parameters, as shown in Figure 6.12C. The transfer function graph contains four nodes, in which case, MARS Vision can analyse the positions of the graph, and attempt to fit it to the ramp shape. It calculates the window as the distance between the second and third nodes, and the level as the value halfway between the second and the third nodes. The colour of the second node is set as the start colour, and the colour of the third node is set as the end colour.

A similar calculation is performed if the graph contains six nodes, in which cases MARS Vision can attempt to fit it to the window shape. Graphs containing a different number of nodes cannot be mapped automatically. In essence, this is a reversal of the procedure used to create a transfer function graph from level and

window settings, as described in section 6.2. The reverse mapping is an optional feature that is never activated automatically.

6.4.2 Summary

The transfer function editor implemented in MARS Vision is based on the GUI concepts described in section 6.3. Minor adjustments have been made for practical reasons, but, overall, no substantial changes were necessary.

6.5 Summary

This chapter has explained the difficulty of transfer function design for energy volumes and the advantages of material volumes, which allow the process of editing transfer functions to be simplified. It has described a novel GUI that is suitable for designing transfer functions for both energy, and material volumes. The key points discussed in this chapter are:

- When visualising energy volumes, users are required to identify the data ranges that correspond to various materials and manually assign colour and opacity to these ranges. The attenuation ranges corresponding to several materials may overlap, in which case they cannot be separated using a transfer function.

- Material decomposition simplifies the representation of the scanned object and places each identified material into a separate volumetric dataset. Therefore, the user only needs to assign distinct colours to different material channels and adjust the visible data ranges and opacity.

- A novel slider-based GUI can be used to design transfer functions for material volumes of a spectral CT dataset. The appearance of each material channel is controlled by a colour and an opacity gradient, created by setting two colours and adjusting the opacity, window, and level parameters. This is similar to the standard approach to CT energy data visualisation that is used in clinical practice. This interface concept can be merged with the traditional GUI for designing transfer functions to create a hybrid interface that is suitable for both energy, and material volumes.

- This concept of the hybrid transfer function editor has been implemented in MARS Vision. Each volume is associated with a separate editor, which maintains the transfer function for that volume.

Chapter VII

Tools for minimising occlusion

This chapter describes three methods for reducing the effects of occlusion during interactive 3D visualisation of spectral CT data. These methods have been developed specifically for working with MARS spectral CT datasets, although they are also applicable to other types of volumetric data.

This chapter is structured as follows:

- Section 7.1 outlines the need for reducing occlusion and clearly visualising selected regions of interest (ROIs).

- Section 7.2 describes the multi-volume threshold intensity projection, a novel technique for visualising multiple channels of a spectral CT dataset. It is an addition to the family of volume raycasting techniques that also includes maximum and minimum intensity projections.

- Section 7.3 explains the functionality of the overlay mode, which brings a single volume to the front, ignoring occlusion entirely. The overlay mode is implemented as an addition to MARS Vision's DVR algorithm.

- Section 7.4 discusses the implementation of the magic lens interaction concept and its application to spectral CT data visualisation. Magic lenses can be used during both 2D, and 3D visualisation.

7.1 Background and motivation

Occlusion is an interesting phenomenon, as it both helps and hinders 3D visualisation. It contributes to the human depth perception, along with other depth cues, such as perspective, motion parallax and stereopsis [237]. It is an obvious visual effect, where the objects located closer to the eye overlap the objects that lie further away from the eye. The same situation occurs in 3D computer graphics

scenes, where occlusion helps create an illusion of depth when viewing an image on a standard 2D monitor.

However, during 3D rendering of volumetric datasets, regions can be totally or partially occluded by other features. This can hinder the process of making an accurate diagnosis, as the user will need to manually move the camera and light position to be able to view all structures of interest. Chapters 4 and 6 have mentioned that occlusion can be minimised by transfer function design. That is, the user can attempt to reduce the opacity of occluding regions by adjusting the transfer function.

However, this is only a partial solution, as occlusion is a view-dependent problem [100]. This means that ROIs may only appear to be obstructed from certain camera positions, as illustrated in Figure 7.1. Further, changing the view angle is not guaranteed to provide a solution. For example, the calcium volume in Figure 7.2 is positioned between two structures from the lipid volume, and will be occluded from any point of view.

Therefore, we can broadly group the effects of occlusion into desirable (enhanced depth perception) and undesirable (obstruction of ROIs). If we consider multi-volume visualisation, then the situation changes substantially, as two types of occlusion are present:

- Intra-volume occlusion, where features of a volume occlude features of the same volume (Figure 7.2A).

- Inter-volume occlusion, where features of a volume occlude features of a different volume (Figure 7.2B).

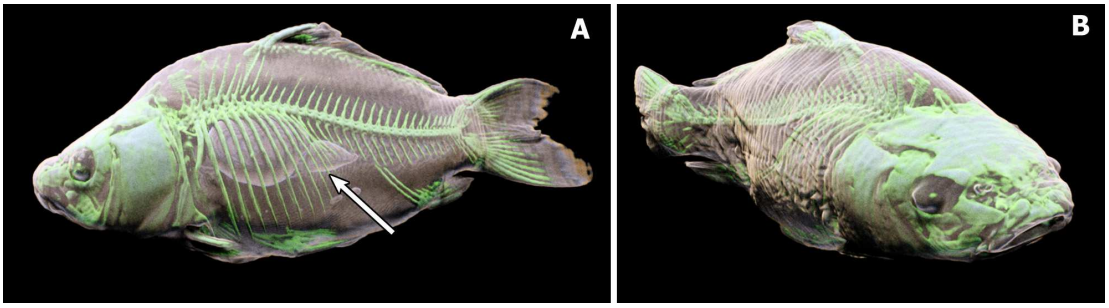


Figure 7.1: The Carp dataset visualised from two different camera positions to show the view dependence of occlusion. The ROI (in this case, the swim bladder), is marked with an arrow.

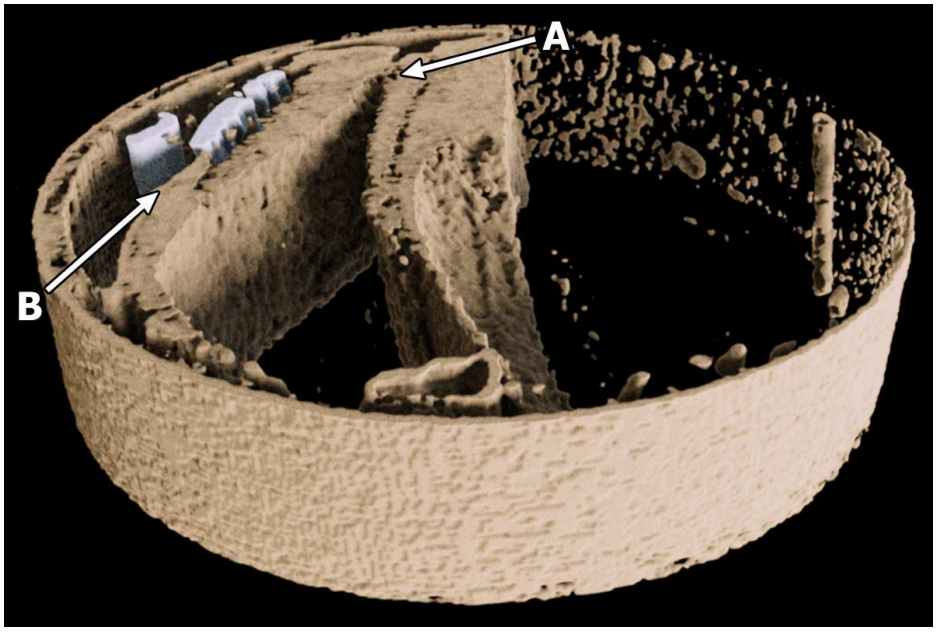


Figure 7.2: 3D rendering of the Meat1127 dataset (section 9.3.1). **A**: intra-volume occlusion. A part of the lipid volume (beige) is occluding another part of the lipid volume. **B**: inter-volume occlusion. The lipid volume is occluding the calcium volume (white). From the user's point of view, this distinction is irrelevant, as there is no visible difference.

From the point of view of the end-user of spectral CT data visualisation software, both types of occlusion are indistinguishable; however, identifying this difference allows us to develop special tools to reduce its undesirable effects. In particular, the segmentation performed by the MD step of the MARS toolchain (section 3.2.2) separates different materials into independent volumetric datasets and presents an ideal opportunity for the development of tools to reduce inter-volume occlusion.

In summary, occlusion is a useful and desirable depth cue that should not be removed entirely. Instead, the known properties of spectral CT datasets should be exploited to reduce inter-volume occlusion and preserve the context.

7.2 Multi-volume threshold intensity projection

This section describes the threshold intensity projection (TIP), a novel visualisation technique that extends the traditional concepts of maximum and minimum intensity projection (MIP and MinIP). Unlike the DVR algorithm described in

Chapter 5, TIP is based on the standard technique of volume raycasting. TIP allows for simultaneous rendering of multiple volumes, where the appearance of each volume is controlled by a single colour parameter, and by the threshold and window settings.

7.2.1 MIP and MinIP

MIP are MinIP are very similar volume visualisation techniques. Both use volume raycasting to sample inside the volume and employ a fixed greyscale colour scheme where lighter colours correspond to regions of high attenuation and darker colours correspond to regions of low attenuation.

MIP shows highly-attenuating structures or materials. Each ray, traversing the volume, only returns the largest voxel value that it has encountered. In clinical CT imaging, these values usually correspond to bones, metal implants, or contrast agents. MIP can be used for displaying small or thin features containing a contrast agent, and is particularly useful for making blood vessels stand out [202]. MIP clearly displays these structures by removing occluding soft tissues, [203], which have lower attenuation values and are therefore not shown. Figure 7.3A illustrates the use of MIP to visualise a conventional CT dataset.

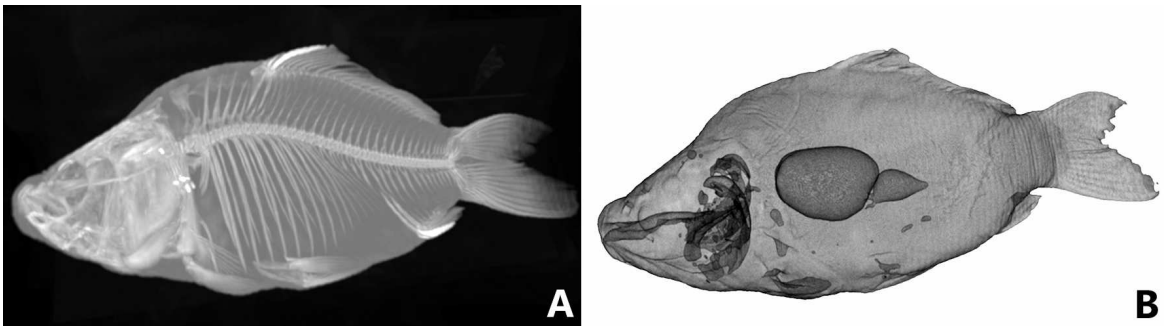


Figure 7.3: Visualisation of the Carp dataset using MIP (**A**) and MinIP (**B**).

MinIP is the opposite of MIP: each ray finds the lowest data value along its path. Usually, this value corresponds to air, which makes MinIP useful for visualising lungs in clinical CT imaging [204, 205]. Figure 7.3B illustrates the use of MinIP. Low-density organs (in particular, the swim bladder of the carp) are clearly visible.

The algorithms for MIP and MinIP are nearly identical:

1. Cast a sampling ray into the volume from each image pixel.

2. Iterate over each ray, sampling the volume at every step.

3. (MIP) Compare the sample to the largest sampled data value. If the current sample is larger, update the largest sampled value.

3. (MinIP) Compare the sample to the smallest sampled data value. If the current sample is smaller, update the smallest sampled value.

4. Convert the largest or smallest sampled value to a percentage (0% - minimum data value, 100% - maximum data value).

5. Set the final pixel colour using a greyscale colour scheme (0%-black, 100%-white).

A related visualisation technique is the average intensity projection, which shows the average value of all voxels sampled by a ray [276]. The sequence of steps performed by these three projections is illustrated in Figure 7.4.

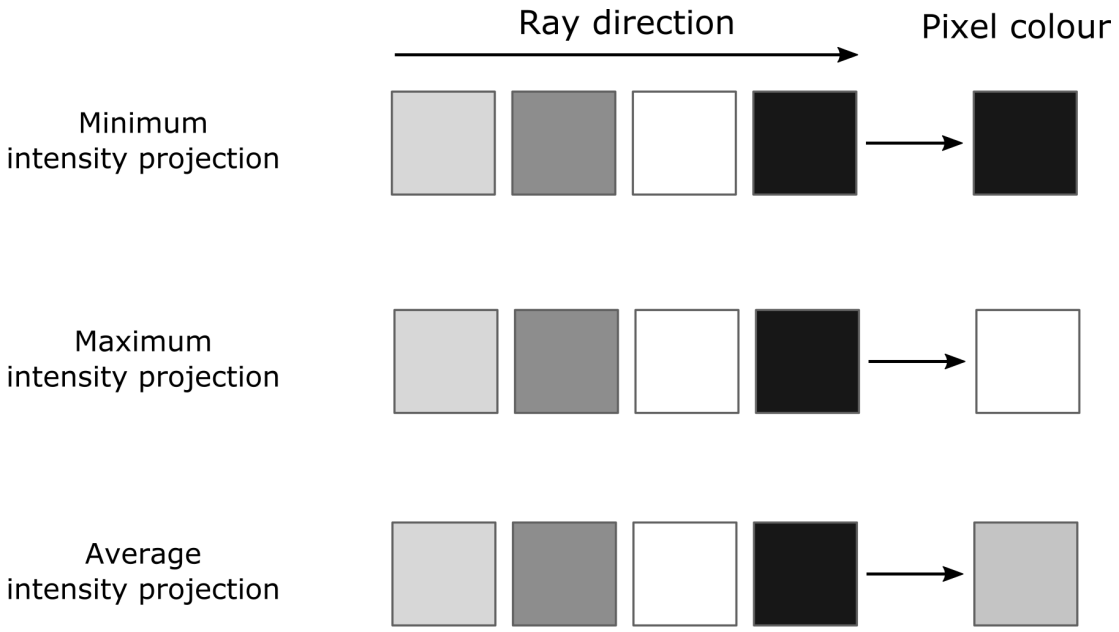


Figure 7.4: Selection of the pixel colour by a sampling ray in three common intensity projections.

These projections are generic volume visualisation techniques and can therefore be used to visualise individual energy or material volumes of a spectral CT dataset. However, their utility is limited, as they do not offer precise control over the visible data ranges: MIP always shows the largest data value sampled by a ray, MinIP always shows the smallest value, and average intensity projection always shows the average of all samples.

Modified versions of maximum and minimum intensity projections have been proposed in the past. For example, the concept of a sliding window may be used: several slices of a volume are grouped into a slab, and minimum or maximum values are found inside it [205, 277]. This restricts MIP or MinIP to a certain region of the volume, which eliminates the structures outside of the ROI.

Another modification to MIP, proposed by Sato et al. [278], is called local maximum intensity projection (LMIP). It finds the first local maximum value along the ray that is above a certain user-defined threshold. Once the local maximum is found, the ray terminates. Therefore, this projection does not necessarily find the largest value along the ray; only the first dense structure is shown, even if denser structures are present behind it. This may, in some cases, enhance the visibility of overlapping structures such as blood vessels. This is shown in Figure 7.5D, where LMIP conveys the geometric information better than the alternatives.

This projection, as well as all other proposed modifications, use the same greyscale colour scheme as the original MIP. This leads to a question: what if this concept was taken one step further, and both the threshold, and the colour scheme were not fixed, but set interactively by the user?

This is the motivation for developing the threshold intensity projection (TIP). To my knowledge, it is the first intensity projection to support simultaneous visualisation of multiple volumes, include a customisable colour scheme, and allow the user to precisely control the visible data range.

7.2.2 Basic single-volume TIP algorithm

This section explains the concept of TIP by describing the simplest possible TIP algorithm for rendering a single volume. In brief, TIP sets the colour calculated by each sampling ray based on the proximity, or distance, of sampled values to a user-defined threshold.

Figure 7.6 shows the sequence of steps performed by the TIP algorithm. First,

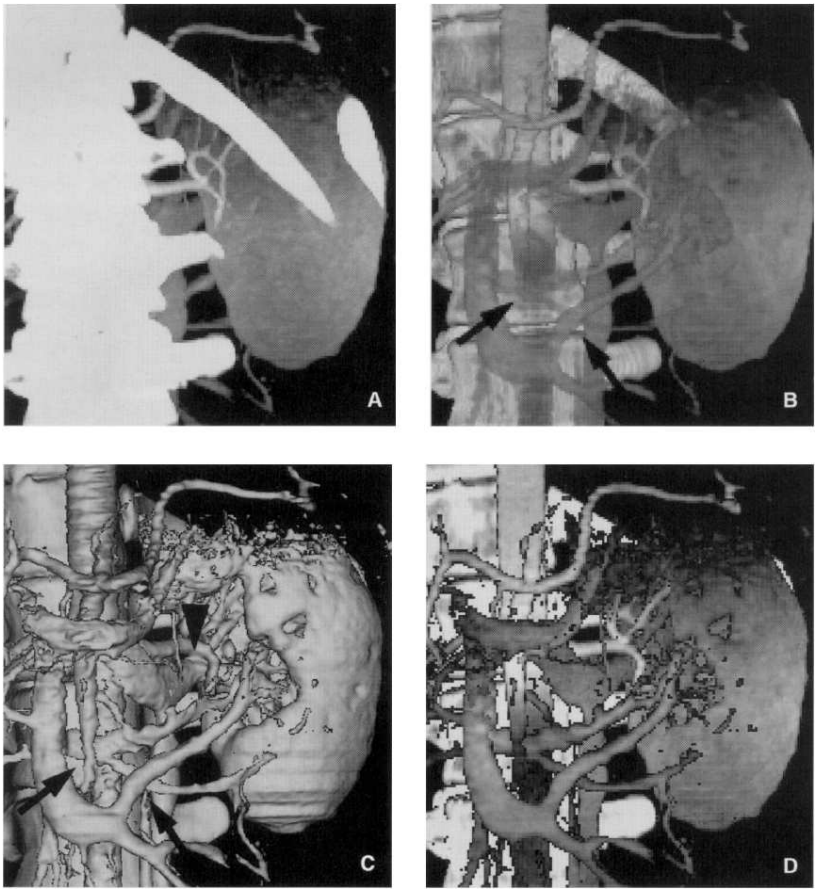


Figure 7.5: Visualisation of a CT angiography dataset of a kidney. **A**: MIP. **B**: volume rendering. **C**: Mesh rendering/surface visualisation. **D**: LMIP. Image from Sato et al., Local Maximum Intensity Projection (LMIP): A New Rendering Method for Vascular Visualization Journal of computer assisted tomography, LWW, 1998, 22, 912-917. Used with permission from Wolters Kluwer Health, Inc.

the user sets a threshold value and a colour to be associated with it. Next, a ray is cast into the volume from every image pixel. The algorithm for ray casting and volume data sampling performed by TIP is identical to the standard MIP, or to ray-marching DVR. Each ray steps through the volume, samples at every step, and stores the closest value to the threshold that it has encountered. This value, or the "proximity to the threshold", is stored as a percentage and calculated according to the following formula:

$$\text{proximity } (\%) = (1 - |\text{sample - threshold}|) \times 100 \tag{7.1}$$

where all data values are expressed as decimal fractions between 0 (the min-

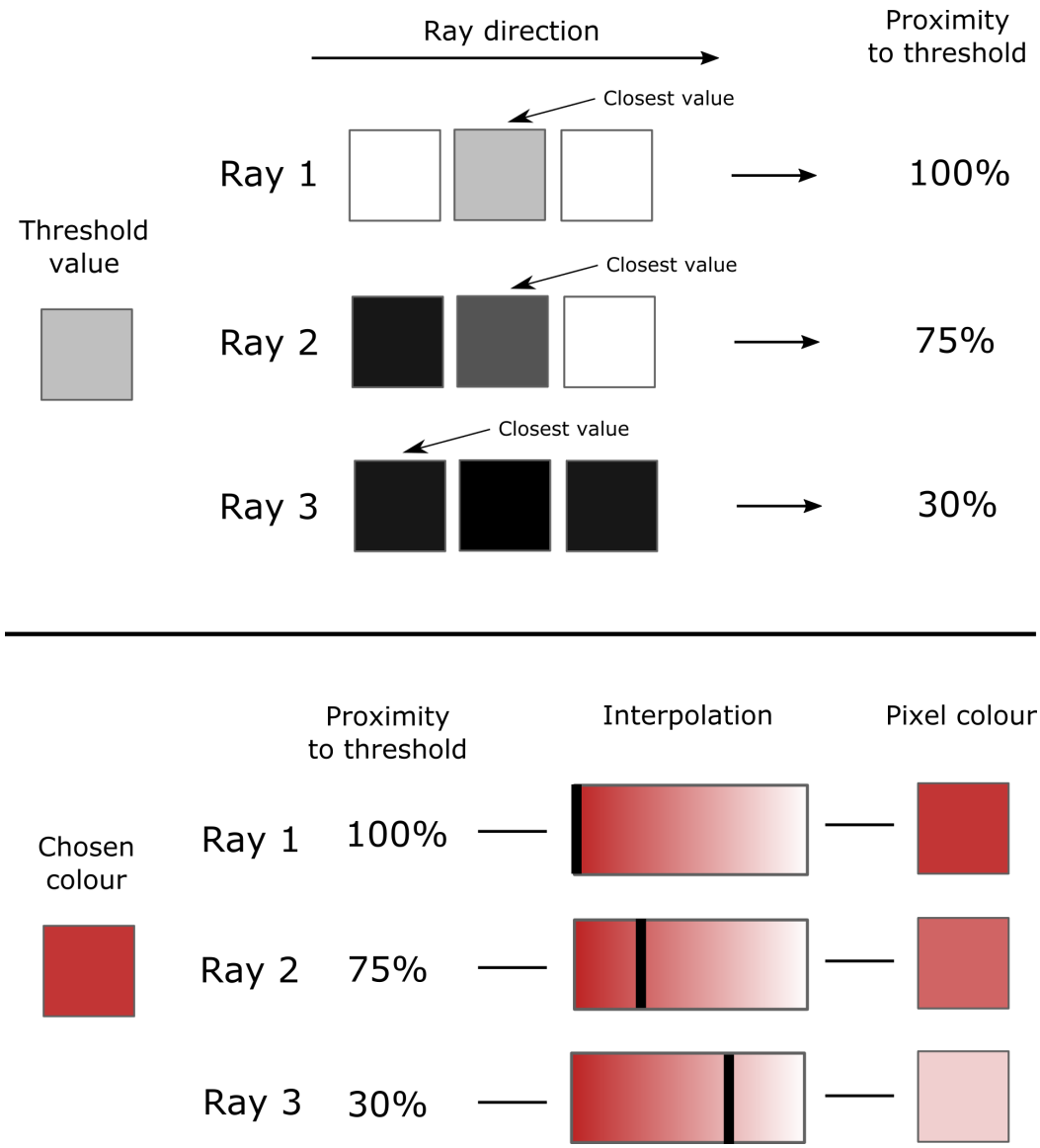


Figure 7.6: The TIP algorithm for a single volume. Top: the first step. The closest value to the chosen threshold is found and stored as a percentage. Bottom: the second step. The final pixel colour for a ray is calculated by interpolating between the chosen colour and white, based on the proximity calculated during the first step.

imum data value) and 1 (the maximum data value). When the rays terminate (leave the dataset's bounding box), the final proximity percentages are converted to colours. This step linearly interpolates between the chosen colour and white, using the calculated percentage difference as the ratio.

Figure 7.7 shows the use of basic TIP to visualise a conventional CT dataset. Changing the threshold reveals different structures inside the body. However, at

lower thresholds (for example, 1000 and 1500 HU), most structures are assigned a very similar shade of colour and become indistinct.

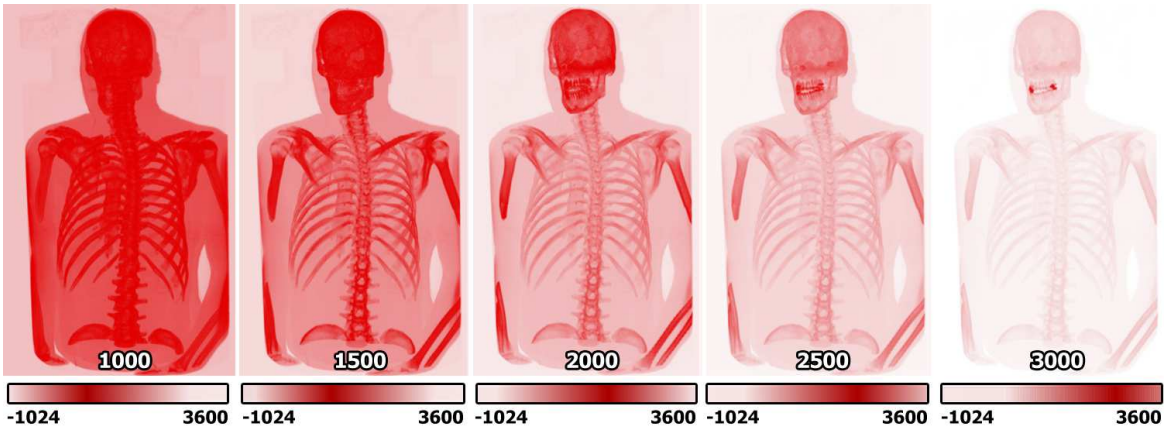


Figure 7.7: Basic TIP rendering of the Visible Human Male dataset. Threshold values are in Hounsfield Units. The colour gradients created by interpolating between the chosen colour and white are shown below the images.

In this example, the reason is that the colour gradient automatically created by TIP changes too slowly to show the small differences in the radiodensity of soft tissue (usually between 40 and 80 HU [279]). This can be seen in the colour bars under the images in Figure 7.7.

However, this problem is not restricted to standard CT datasets (energy volumes), and will always occur when different tissues or materials occupy similar data ranges. Therefore, the threshold parameter alone is not sufficient for visually separating such ranges, and a second parameter must be added to control the slope of the colour gradient. This parameter is called "window", following the convention used to describe the transfer function parameters in Chapter 6.

7.2.3 Standard multi-volume TIP algorithm

The standard TIP algorithm implemented in MARS Vision supports multiple volumes and adds a "window" parameter that controls the steepness of the colour gradient.

There are several methods of controlling the window size. MARS Vision uses an arbitrary window value that modifies the proximity according to the following

formula:

$$\text{proximity } (\%) = (1 - |\text{sample - threshold}|^{\frac{1}{\text{window}}}) \times 100 \tag{7.2}$$

This algorithm uses the window parameter to scale the difference between the sample and the threshold. By default, the window is 1, which is identical to the basic TIP algorithm. Larger window parameters greatly reduce the proximity to the threshold.

The following example can be used to illustrate the effect of window size. Suppose that the volume data ranges from 0.0 to 1.0, the sample is 0.45 (45%), and the threshold is 0.2 (20%). If the window is not adjusted, then the difference is 25% and therefore the proximity is 75%. If the window is set to 10, then the proximity is:

$$(1 - |0.45 - 0.2|^{\frac{1}{10}}) \times 100 = 12.94\% \tag{7.3}$$

This means that the final interpolated colour is now around 13% away from white, as opposed to the original 75%. The window, or the data range over which the chosen colour and the background colour are interpolated, is now much narrower. This allows the user to find specific tissues or materials, while hiding all other tissues, as shown in Figure 7.8. Figure 7.9 shows the effect of the window parameter on 5 different thresholds. Values above 10 result in a significant reduction of the displayed data range.

Finally, extending TIP to multiple volumes (or multiple different thresholds for a single volume) is a trivial task that requires minor alterations to the sampling step and the addition of a second colour blending step:

1. The user sets a threshold, a window and a colour for each volume. Appropriate threshold and window settings need to found manually, but, unlike in traditional transfer function design, there are only two parameters to adjust.

2. Rays are cast into the volume from each image pixel, as in the basic TIP.

3. Each ray iterates through the dataset, sampling all volumes at every step. The proximity to the threshold is maintained for each volume separately.

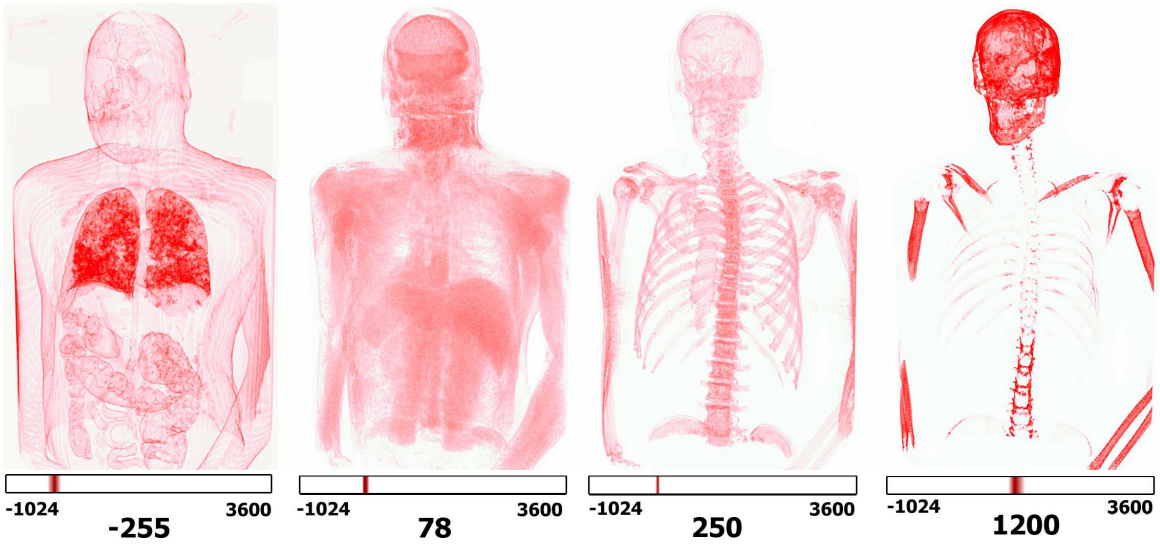


Figure 7.8: TIP rendering of the Visible Human Male dataset using narrow window settings. Threshold values are in Hounsfield Units. The approximate colour gradients created by interpolating between the chosen colour and white are shown below the images. Due to the small size of the window, only a very small attenuation range is shown. This allows the user to precisely target a particular tissue type.

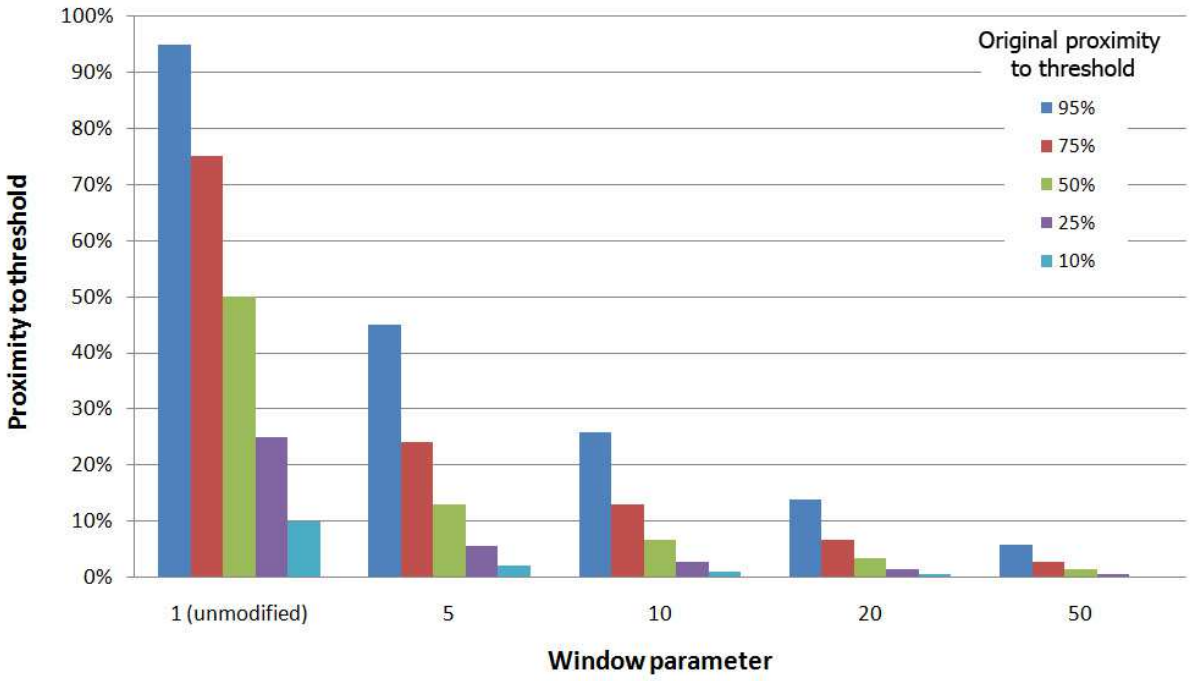


Figure 7.9: The effect of the window parameter on the proximity to the threshold.

4. When the ray terminates, all proximity values are converted into colours, according to the formula in Equation 7.2.

5. All colours are blended together in the CIE XYZ 1931 colour space. The colours are multiplied together, which preserves the normalised range (0-1) for each component (X, Y, and Z). The colour black is represented by (0, 0, 0) in the XYZ colour space. Therefore, multiplication draws all components towards 0, which makes the areas where several volumes overlap darker, and improves the contrast in the image. This is also the motivation for using white as the background colour in the algorithm.

7.2.4 Use of TIP

This section briefly discusses the use of TIP for visualising occluded ROIs, while further examples of the possible use cases can be found in Chapter 9. TIP is used when several overlapping volumes need to be displayed simultaneously to provide context. In this case, most volumes can be visualised as faint outlines (by selecting narrow window parameters).

Figure 7.10 provides three examples of the use of TIP. In this figure, the transfer functions used in DVR attempt to imitate the translucency achieved by TIP. The data for all volumes is stored in relative concentration units (see section 9.0.1 for a detailed explanation of this data format).

The TIP rendering in Figure 7.10A shows calcium in green and iodine in red. DVR shows calcium as a translucent white isosurface and iodine in orange. Some regions containing iodine are occluded by the calcium volume (marked with white arrows). TIP displays the outline of the iodine volume and reduces the effects of occlusion while retaining the context, which is provided by the calcium volume.

The TIP rendering in Figure 7.10B TIP shows calcium in blue, water in green, and lipid in red. DVR shows calcium in white, water in red and lipid in beige. DVR does not show the bottom part of the calcium volume, as it is occluded by the lipid volume. In contrast, TIP shows the outline of the entire calcium volume. The marbling patterns in the lipid volume (marked by orange arrows) are shown by both techniques, although the outline of the lipid in the middle (which is curved) is clearly shown with TIP. DVR only shows the top surface of the lipid volume, even though a narrow window-shaped transfer function is used in an attempt to produce a similar outline (see the colour bar for the lipid volume).

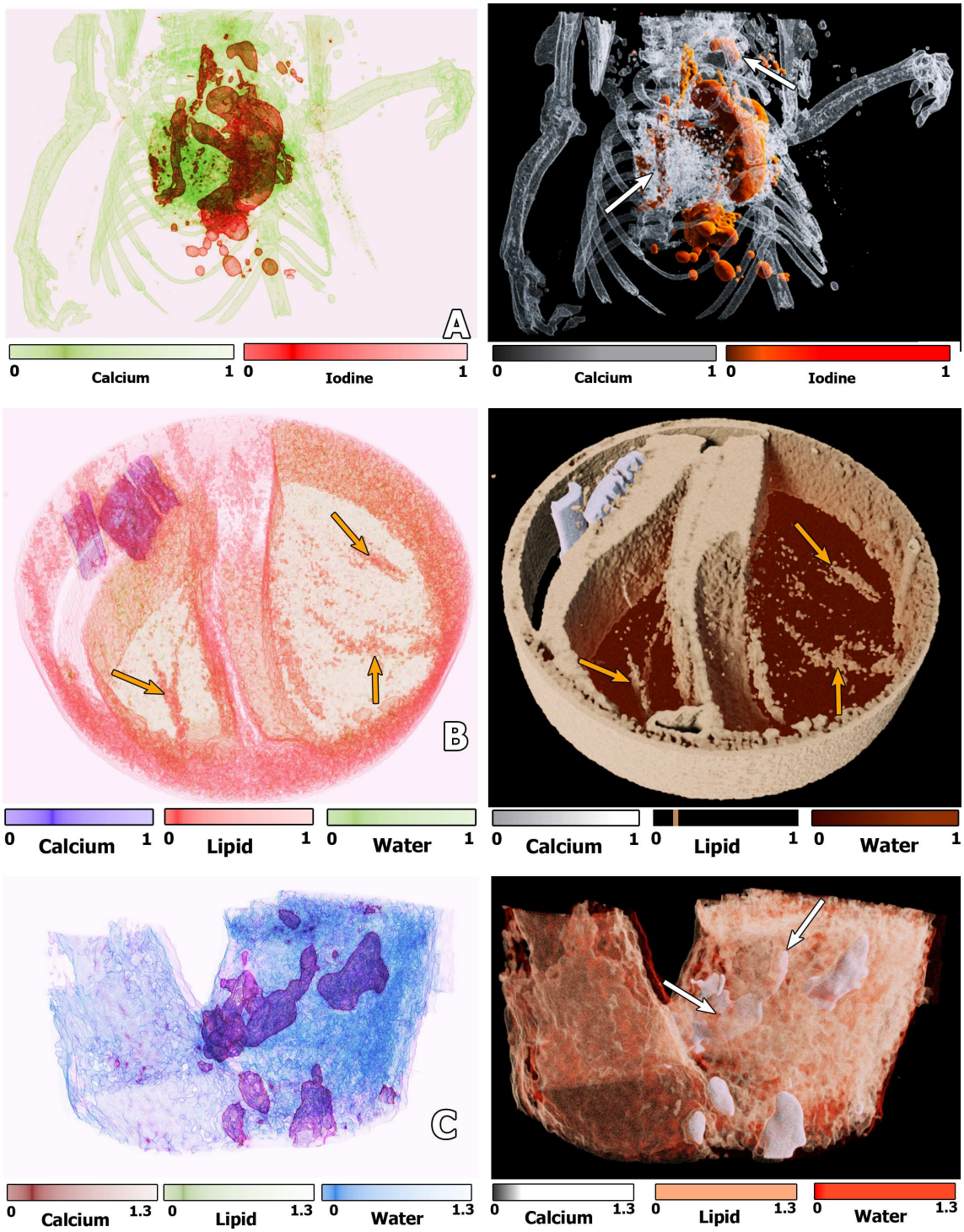


Figure 7.10: TIP (left) and DVR (right) rendering of three MARS spectral CT datasets. **A**: Mouse12 (section 9.6.1). **B**: Meat1127 (section 9.3.1). **C**: Plaque72 (section 9.2.1). See text for further details.

The DVR in Figure 7.10C uses the same colour scheme as Figure 7.10B, but makes the lipid and water volumes translucent to reduce occlusion. DVR does not clearly show the calcifications in the artery, as they are occluded by both the lipid and the water volumes. The opacity of the lipid and water volumes has been adjusted to improve the visibility, but occlusion still poses a problem. In contrast, TIP shows the outline of the calcifications, which appears as purple due to blending with the blue colour assigned to the lipid volume.

Overall, Figure 7.10 shows that DVR provides superior visual quality, but is also more seriously affected by occlusion. TIP can be configured to retain the context and to clearly display the outlines of ROIs.

The most serious problem with TIP is the loss of depth information. Like other intensity projections, it does not provide sufficient depth cues to the user. However, it is possible to visualise the precise location of features by using other cues, primarily motion parallax. Another option is to use stereoscopic 3D rendering, as discussed in section 10.2.3.1.

A second, minor, problem is the lack of support for complex colour gradients that can be used to separate features during DVR visualisation. This is intentional, as TIP is designed to:

1. Clearly show the position of ROIs.

2. Minimise occlusion of ROIs.

3. Accomplish this using the smallest possible number of user-configurable parameters. The appearance of TIP visualisation is designed to be simple to adjust, as only three parameters need to be set for each volume (threshold, window, and a single colour). TIP requires neither opacity, nor lighting configuration. In contrast, DVR requires transfer functions to be set, opacity to be configured, and lighting to be adjusted.

In comparison, DVR is a general-purpose visualisation technique, which uses arbitrary mapping of volume data to colour and opacity to achieve a wide variety of visual effects. However, the desired appearance may be hard to achieve. TIP, on the other hand, is intended to be simple to use and configure.

Finally, consider the differences shown in Figure 7.11. Here, DVR and TIP have been configured to show the Carp's skin, swim bladder, and respiratory system.

DVR shows various features (such as the skin and the swim bladder) overlapping in several places, while TIP only shows a single surface at each pixel. This is a significant difference from window- or triangle-shaped transfer functions, where multiple translucent surfaces may overlap. As shown in this figure, TIP can increase the visible surface detail.

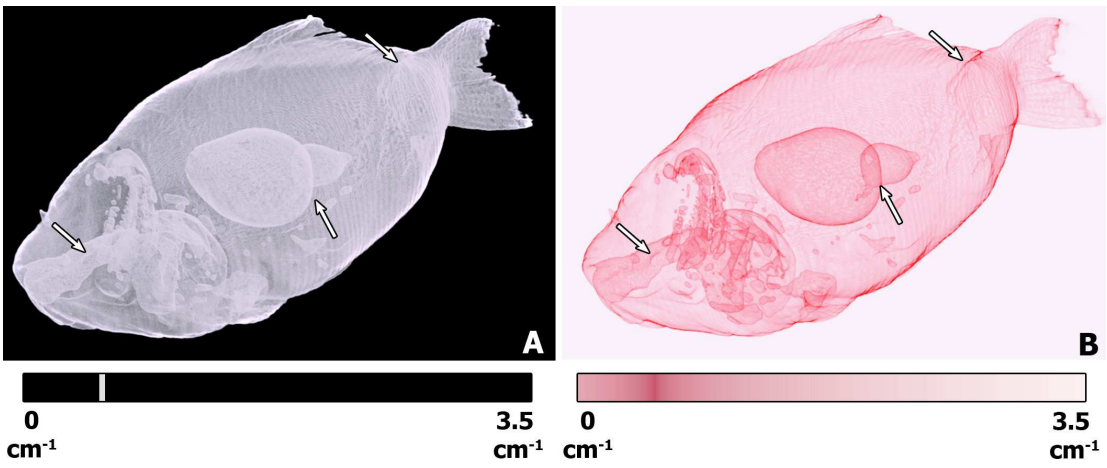


Figure 7.11: DVR (**A**) and TIP (**B**) visualisation of the Carp dataset. Areas where the difference between these two techniques is most obvious are marked by arrows.

7.2.5 Emphasis mode

TIP does not suffer from occlusion, in the standard definition of the term. However, the visibility of ROIs may be reduced as a result of blending of multiple colours, as illustrated in Figure 7.12A. Here, the colour for iodine (red) blends with the colours for calcium (green) and barium (blue). As a result, the region marked by an arrow appears dark, and detail is lost.

The emphasis mode effectively "fades" other volumes to clearly display the chosen volume. This is done by modifying the contribution, or weight, of each colour during the colour blending stage. By default, the weight is 1.5, which means that the proximity to the threshold for all volumes, except the emphasised one, is calculated according to the following formula:

$$\text{proximity } (\%) = \frac{(1 - |\text{sample - threshold}|^{\frac{1}{\text{window}}}) \times 100}{1.5} \tag{7.4}$$

This formula shows that the weight of 1.5 reduces the proximity to the threshold

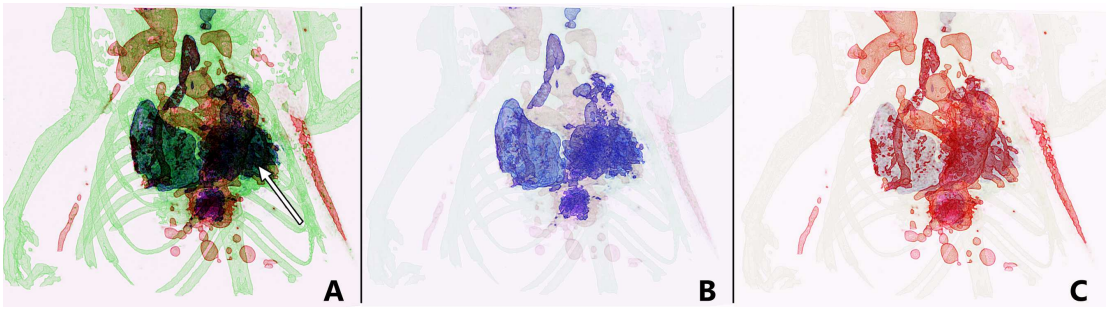


Figure 7.12: TIP visualisation of the Mouse12 dataset (section 9.6.1) using the standard TIP mode (**A**) and the emphasis mode, with the barium volume being emphasised in **B** and the iodine volume being emphasised in **C**.

by 50%. This value was found to provide a good balance between the visibility of the emphasised volume and the other volumes. Lower weights do not clearly emphasise the chosen volume, while higher weights can hide the other volumes almost completely (by bringing the proximity to the threshold close to 0%, and drawing the colour close to white).

The standard TIP colour blending algorithm blends the colours calculated for all volumes using equal weights. The emphasis mode reduces the weight assigned to all volumes except the chosen emphasis volume. The results are shown in Figure 7.12B and C.

7.2.6 GUI for controlling parameters

This section briefly describes the GUI elements that are used to adjust the TIP settings for multiple volumes. The GUI implemented in MARS Vision is shown in Figure 7.13A. Note that a GUI widget for controlling the settings for a single threshold resembles the simple form of MARS Vision's hybrid transfer function editor (section 6.3). The widgets for controlling multiple thresholds are laid out in a list, similar to the layout proposed for accommodating multiple hybrid transfer function editors.

Each widget shown in Figure 7.13A allows the user to set the threshold (1), window (2), and colour (3) settings for a particular volume. The volume can also be changed (4), which allows the user to set several different thresholds for one volume. Finally, a particular volume can be emphasised (5), or hidden (6).

Figure 7.13B illustrates the result of setting the thresholds as shown in A. Here, three of the four volumes of the Mouse0 dataset are shown simultaneously. Gold

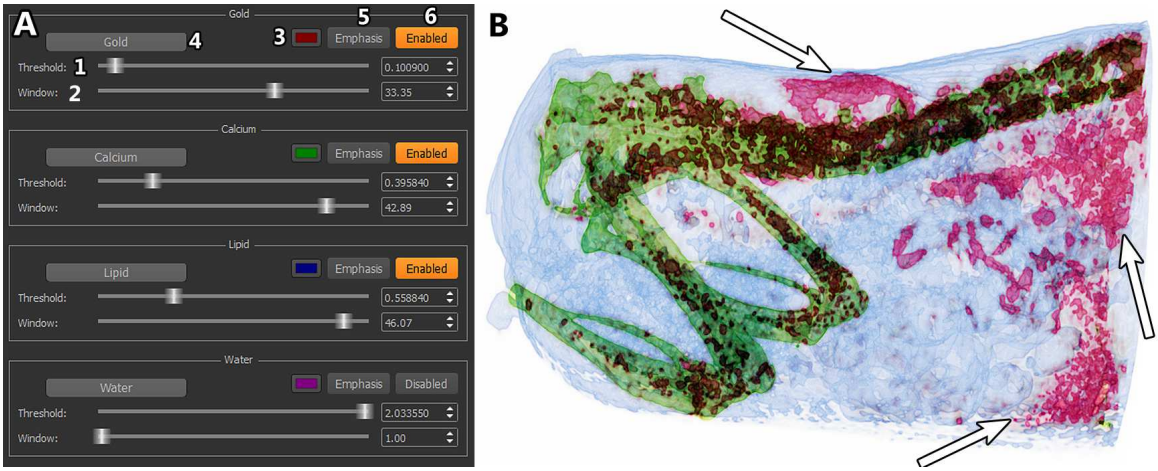


Figure 7.13: TIP visualisation of the Mouse0 dataset (section 9.6.2). **A**: the GUI for setting multiple TIP thresholds and colours. **B**: the resulting TIP visualisation of three material channels. The gold that has accumulated inside the liver and inside the tumour implanted under the skin of the mouse is marked by arrows.

is shown in red, calcium in dark green, and water in blue. The lipid channel is disabled, as it is noisy and does not contain much useful information. The location of gold is clearly shown along with the context.

7.2.7 Performance

Table 7.1 shows the performance of MARS Vision's implementation of TIP. As described in section 5.3, MARS Vision's DVR algorithm progressively refines the image with every iteration. Therefore, its performance is measured in iterations per second. In contrast, TIP employs the standard ray marching technique, which renders the image fully before displaying it (no progressive refinement); its performance is measured in frames per second. Therefore, a direct comparison of the performance of DVR and TIP cannot be easily made.

As discussed in sections 5.3 and 5.5, DVR requires around 5 iterations to render recognisable structures, around 20 iterations to generate an image of acceptable quality, and around 100 iterations to generate a very high-quality image. On the other hand, TIP only requires a single iteration, as there is no progressive image refinement. Therefore, even this basic, unoptimised implementation of TIP is already significantly faster than MARS Vision's DVR algorithm.

One significant difference between TIP and DVR must be discussed. In MARS Vision's DVR algorithm, each sampling ray attempts to reach a certain target

Table 7.1: The performance of MARS Vision's DVR and TIP algorithms. Images are rendered at 800×600, with a stationary camera. The rendering parameters used for DVR are described in section 5.5. The performance of DVR is measured in iterations per second, while the performance of TIP is measured in frames per second.

Dataset	Dimensions, volumes	DVR	TIP
Mouse12	$338 \times 440 \times 257, 4$	7.41	25.28
Spectral Phantom	$246 \times 246 \times 16, 5$	31.37	123.48
Meat1127	$436 \times 436 \times 126, 3$	38.41	49.22
Mouse0	$414 \times 414 \times 451, 4$	31.90	7.30
Knee Cartilage	$544 \times 460 \times 587, 2$	21.85	10.07
Plaque108	$384 \times 384 \times 288, 3$	30.64	24.45
Plaque72	$583 \times 583 \times 99, 4$	30.57	24.48

opacity and terminates if that target has been reached (see section 5.3.3). Therefore, the performance of DVR varies depending on the transfer function settings: high opacity settings, such as those used for the Knee Cartilage dataset (shown in Figure 5.20) require few samples, as most rays terminate quickly. Low opacity settings require more samples. This reduces the performance as rays take longer to find their target opacity values.

In contrast, the performance of TIP is much more consistent and predictable. A ray always starts sampling at the point of entry into the dataset's bounding box and always terminates at the point of exit from the bounding box. The calculation in Equation 7.2 is always performed at each step along the ray, regardless of the values of the threshold and window parameters; no opacity calculations are required.

Therefore, the threshold and window settings only affect the appearance, but not the performance of TIP. In comparison, the transfer functions used in DVR affect both the performance of the algorithm and the appearance of the image displayed to the user.

In conclusion, the proposed implementation of TIP is not only interactive, but is also faster than DVR. However, this implementation has not been optimised, and further work on acceleration techniques for TIP will be beneficial in the future. For example, a standard octree (section 5.5.2) that stores the minimum, maximum, or average value of a block of voxels, may be modified to support TIP. This can be done by storing the proximity to the threshold for every block of voxels in a

volume. The rendering algorithm can skip all blocks where the proximity values are smaller than a certain limit (perhaps around 1-5%, but the exact limit will need to be determined experimentally).

7.2.8 Disadvantages

7.2.9 Summary

Multi-volume threshold intensity projection is a novel visualisation technique that extends the concepts of maximum and minimum intensity projection. It can be used to simultaneously render multiple volumes while minimising the effects of occlusion. The user assigns a colour to each volume and uses the threshold and window parameters to control its appearance and set the visible data range. During spectral CT data visualisation, TIP allows the user to precisely target particular attenuation ranges (in energy volumes) or concentration ranges (in material volumes). Other currently-known intensity projections, such as MinIP or MIP, do not allow this degree of precision.

TIP can also be used to aid in transfer function design, as it allows the user to identify thresholds that match transition points. This is because the structure suddenly undergoes significant change that generally corresponds to material boundaries.

7.3 Overlay mode

The overlay mode is a visualisation technique that can be used to improve the visibility of features during multi-volume rendering. One volume is rendered as though it is located in front of all other volumes, which negates inter-volume occlusion (intra-volume occlusion is still present). This mode is implemented as an addition to MARS Vision's DVR algorithm, although the concept of an overlay volume is generic and can be implemented in other multi-volume DVR algorithms. It is similar to known non-photorealistic illustrative rendering techniques, such as those proposed by Rheingans and Ebert [125], Bruckner et al. [280], and especially to the cutaways described by Feiner and Seligmann [281].

Designating a volume as an *overlay* means that its features will not be occluded by any feature from other volumes, as shown in Figure 7.14. The overlay volume is rendered normally to preserve its structure, but is shown "on top" of other

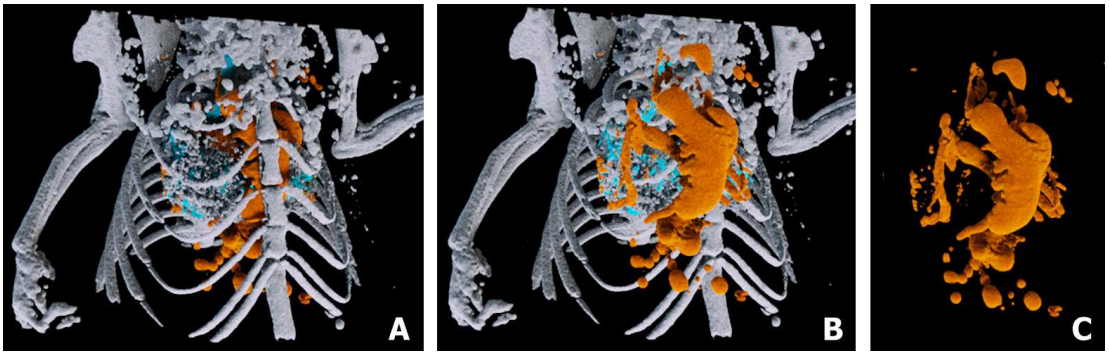


Figure 7.14: Use of the overlay tool to show iodine (orange) inside the Mouse12 dataset (section 9.6.1). **A**: overlay mode off. **B**: overlay mode on. **C**: iodine volume only.

volumes. This can be used to draw the user's attention to the location of potential ROIs inside it.

The overlay mode is designed to be used with material volumes, which already have a degree of separation between them. Another potential use case for this concept is overlaying a PET dataset onto a CT dataset (for PET/CT data visualisation). Finally, overlaying a volume produced by a segmentation routine onto the original, unsegmented volume, is also possible. This section discusses the visualisation of MARS spectral CT datasets using overlays, while the use cases described above may be investigated in the future.

In conclusion, this visualisation technique is only usable if there are significant structural differences between the overlay volume and the other volumes. For example, in Figure 7.14, the calcium volume (grey) is substantially different from the overlaid iodine volume (orange). In this situation, it is clear which volume is overlaying which, and where the iodine is located. On the other hand, energy volumes are very similar structurally, so overlaying one on top of another is unlikely to yield a clear and meaningful visualisation.

7.3.1 Implementation and use

The implementation of the overlay mode in MARS Vision extends the basic DVR algorithm described in section 5.3. This allows the overlay mode to benefit from the depth cues provided by the illumination used in DVR. In contrast, MIP or TIP do not calculate illumination, and thus lose a valuable source of depth information.

Algorithm 4 shows the necessary changes to MARS Vision's DVR. The scat-

tering point selection step (section 5.3.3) and the illumination calculation step (section 5.3.4) must be modified. If an overlay volume is selected, then the algorithm proceeds as follows:

- If a ray finds a valid scattering point inside the overlay volume, then no other volumes are sampled or shaded during the illumination step.

- If a ray does not find a valid scattering point inside the overlay volume, then rendering continues normally.

This algorithm preserves the context. Consider Figure 7.14C, where the material of interest is displayed separately. The context is lost, as there are no cues to indicate where these materials are located, and how they are distributed relative to other materials. Using the overlay mode displays both the context and the chosen material, as shown in Figure 7.14B.

In general, the overlay mode can be used when a particular volume of interest is small, but occluded by several other volumes, and standard methods of reducing occlusion are not appropriate. For example, clipping planes may be ineffective if the geometry of the occluding structures is too complex, while adjusting the transparency of these structures may lead to the loss of context.

7.3.2 Advantages, disadvantages, and future work

The primary advantage of the overlay mode is that it ensures that the chosen volume is never occluded by any other volume, while the other techniques proposed in this chapter (magic lenses and TIP) cannot guarantee this. In addition, it requires no configuration beyond selecting the overlay volume.

However, the overlay mode also distorts the apparent position of the overlaid volume relative to the other volumes by providing conflicting depth cues. A similar problem also affects intensity projection techniques: for example, MIP always bring the densest structures to the front regardless of their real location, while the overlay mode always brings a chosen volume to the front.

This problem can be mitigated by interacting with the visualisation: viewing the dataset from different angles allows the observer to use motion parallax (which is a very important and effective depth cue for the human visual system [282, 283]) to improve depth perception, as shown in Figure 7.15. In addition, the overlay

Algorithm 4 Modifications to the standard MARS Vision's DVR algorithm (Algorithm 1) to enable overlay mode. The modifications are shown in red.

Data:
Ray R cast from pixel $P_{x,y}$ with origin R_o and direction R_d, n volumes $V_{1...n}$, n transfer functions $T_{1...n}$, overlay volume index I

Result: Colour value C for ray R

initialization:
temporary colour TC, temporary opacity TO
scattering location $S \leftarrow \infty$
total opacity $O \leftarrow 0$

attempt to find a scattering point in the overlay volume:
use Woodcock tracking to find S in V_I
if $S = \infty$ **then**
  find S using the standard scattering point selection algorithm
end

update depth buffer for P, using S

lighting and shading calculations:
if $S \neq \infty$ **then**
  if valid sample in overlay volume **then**
    $O \leftarrow$ calculate opacity at position S in V_I using T_I
    $C \leftarrow$ calculate lighting for position S in V_i using T_I
    $C \leftarrow C \cdot O$ *multiply colour by opacity*
  else
  for $i \leftarrow 1$ **to** n **do**
    $TO \leftarrow$ calculate opacity at position S in V_i using T_i
    $TC \leftarrow$ calculate lighting for position S in V_i using T_i
    $TC \leftarrow TC \cdot TO$ *multiply colour by opacity*
    $O \leftarrow O + TO$ *accumulate opacity*
    $C \leftarrow C + TC$ *accumulate colour*
  end
  opacity correction:
  if $O \neq 0$ **then**
    $C \leftarrow C \div O$
  end
end

save C

mode is never activated without the user's knowledge, which means that the user is always aware that conflicting depth cues may be provided.

Nevertheless, improvements to the overlay mode need to be investigated in the future. For example, the overlay volume may be rendered as a cutaway [281], which would preserve the depth order and reduce the amount of unclear or misleading depth cues. Another potential research direction would be displaying the outline, or contour, of the overlaid volume instead of rendering it fully.

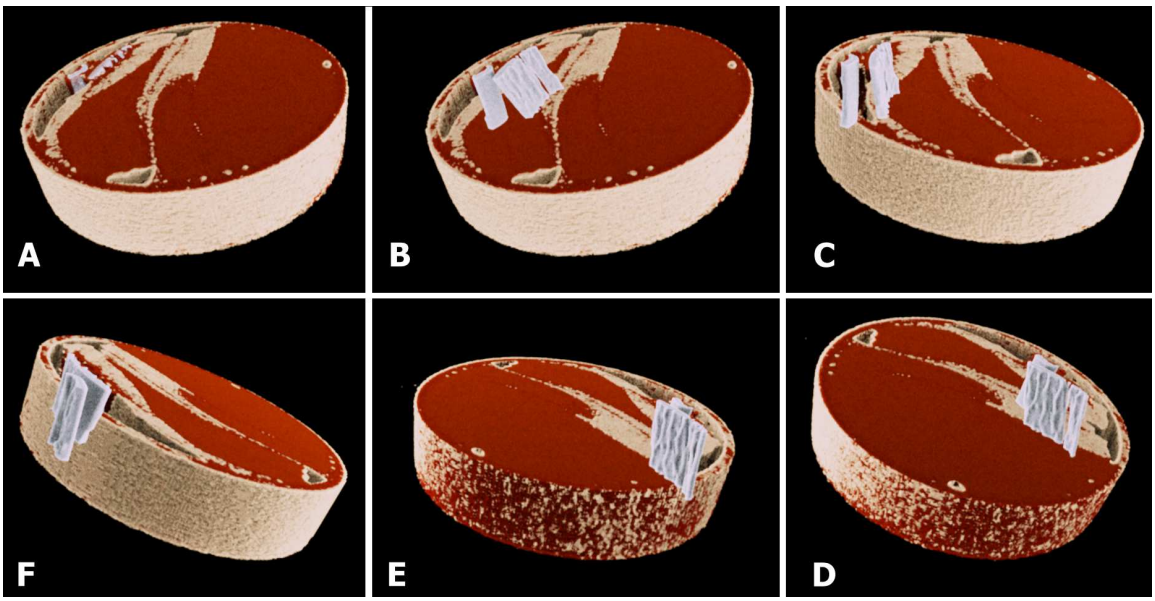


Figure 7.15: Rendering of the Meat1127 dataset (section 9.3.1) from several angles with the overlay mode turned off (**A**) and with the calcium volume set as the overlay (**B-F**).

7.3.3 Summary

The overlay mode brings one volume to the front and displays it without occlusion by other volumes. It is a simple technique that does not require any substantial changes to the rendering pipeline, and can be easily added to a multi-volume DVR algorithm.

7.4 The magic lens as an interaction technique for spectral CT visualisation

This section describes how magic lenses can be used to display an ROI during the visualisation of multiple channels of a spectral CT dataset. It demonstrates

how this interaction technique can be used to enhance both DVR, and 2D slice visualisation.

A magic lens is a movable filter that alters the appearance of data inside it [284]. It has been used for various purposes, such as scientific data visualisation [285], interaction with augmented reality scenes [286] and map navigation [287]. The main advantage of a magic lens is that it allows the user to interactively select a desired ROI and change its appearance, while retaining the visual context. For example, a magic lens moved over a virtual map can be used to magnify its visible features, or to overlay additional information that is normally hidden.

The addition of magic lenses to volume rendering algorithms for medical data visualisation has been described before [288, 289]. Magic lenses were also used as part of augmented reality systems for visualising and interacting with medical datasets [290, 291]. In clinical settings, magic lenses have been used for visualising lung perfusion (blood flow in the lungs) [292], as shown in Figure 7.16.

The review conducted in section 4.3 has found no mention of the magic lens tool being used for spectral CT data visualisation. However, magic lenses and other similar "focus+context" techniques are considered useful for visualising multi-variate data [120, 123, 281] because one data type can be displayed inside the lens, while another data type can be displayed outside. Spectral CT data is a special case of multi-variate data; therefore, the application of magic lenses to spectral CT data visualisation should also be examined.

The magic lens tool is ideally suited for displaying regions of interest inside MARS spectral CT datasets because they have already been partitioned into multiple material volumes. Therefore, the user only needs to select the volumes that should be shown inside the lens, place it, and adjust its size. No other parameters need to be configured (assuming that the colour schemes for the material volumes have already been set).

7.4.1 Use of the magic lens for spectral CT data visualisation

In conventional and spectral CT imaging, the ROI normally includes some feature or material of interest inside the body, while the surrounding tissues or materials comprise the context. Usually, in the datasets acquired by the MARS team, the materials of interest are contrast agents based on elements such as iodine, barium or gold. These materials are embedded inside the tissues that occupy a much

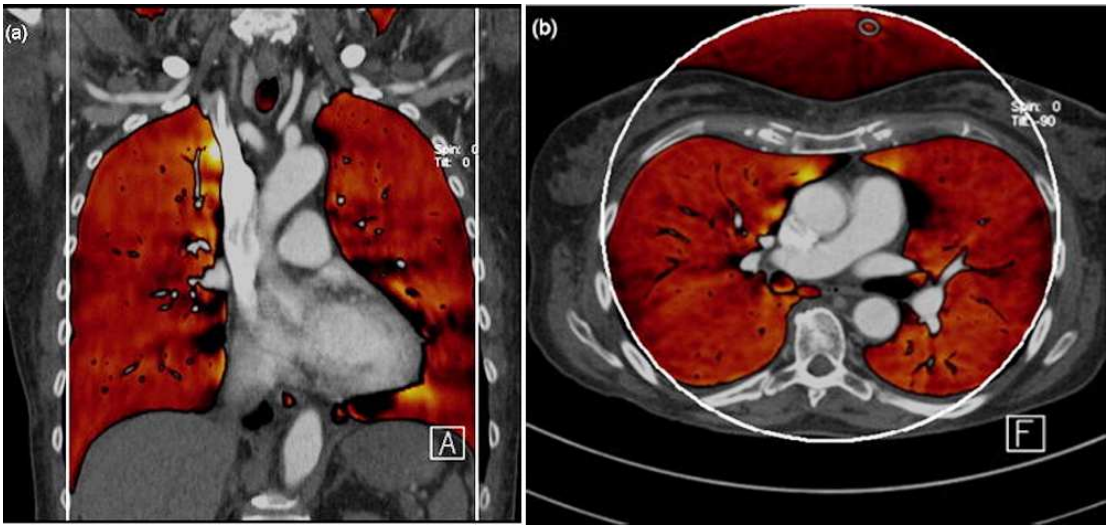


Figure 7.16: Use of a movable window to display a perfusion map (yellow/orange/red) inside an ROI during dual-energy CT angiography. The CT image is used as a background. Figure from: S. Thieme et al. Dual energy CT for the assessment of lung perfusion: correlation to scintigraphy. European journal of radiology, Elsevier, 2008, 68, 369-374. Reproduced with permission from Elsevier B.V.

larger volume of space (for example, soft tissues).

Research has indicated that spectral CT is likely to be used for tasks such as assessing cartilage health, diagnosing unstable atherosclerotic plaque, tumour imaging, diagnosing inflammation and bone densitometry [2]. In most of these cases, the ROI is small compared to the size of the entire dataset.

Figure 7.17A illustrates a typical problem with using 3D visualisation in this situation. In this case, the contrast agent volume (cyan) is displayed alongside other tissues. Occlusion poses a problem, as the ROI containing the contrast agent is located underneath soft tissues and bones.

Figure 7.17B shows that the identified contrast agent may be visualised separately. However, this approach may not provide sufficient context, as the required anatomical information is not present in the contrast agent volume.

Figure 7.17C shows the proposed solution to this problem. The contrast agent volume is displayed separately, but only within a small movable window. This window, referred to as a "magic lens", excludes certain volumes (chosen by the user) from being shown inside the ROI. The ROI can now be displayed clearly, while the rest of the image is unaffected and provides anatomical context. In addition, the lens draws the viewer's attention, which can provide direction to new observers.

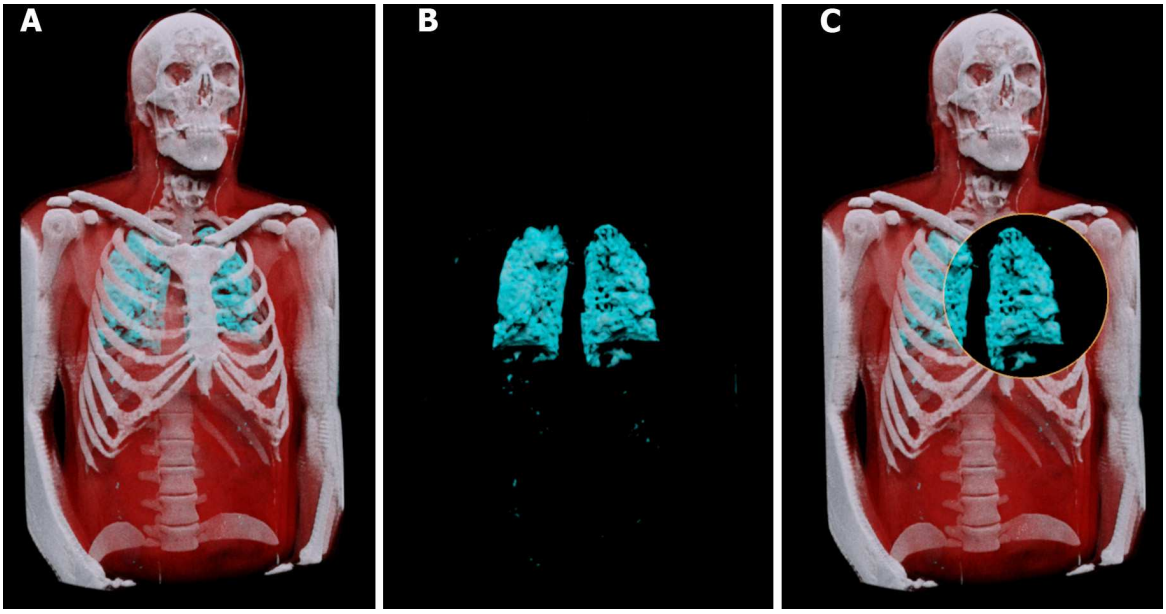


Figure 7.17: Possible future use of the magic lens for visualising a human spectral CT dataset. **A**: soft tissues (red), bones (grey) and a contrast agent (cyan) shown together. **B**: contrast agent only. **C**: a part of the contrast agent volume is shown in a magic lens; the rest of the body is rendered normally to provide context.

The magic lens is a general-purpose focus+context technique, and the nature of the occluding structures, or the ROI, is irrelevant. In the expected common use case of spectral CT in clinical settings, the occluding volumes will contain soft tissues, while the ROI will likely contain some material of interest. If spectral CT is used for other purposes, such as non-destructive testing (which can currently be carried out using conventional CT [293, 294]), then the ROI and the occluding structures will be different, depending on the application.

However, the magic lens tool is not applicable in all situations. It is effective if the ROI occupies a small region of space, such as the example in Figure 7.17. The larger the volume occupied by a material, the less useful a moving window becomes. This is shown in Figure 7.18, where the two volumes occupy roughly equal volumes of space and there is little to no overlap between them. In this case, the calcium volume can be visualised with a magic lens, but it may also be visualised by itself without losing much context.

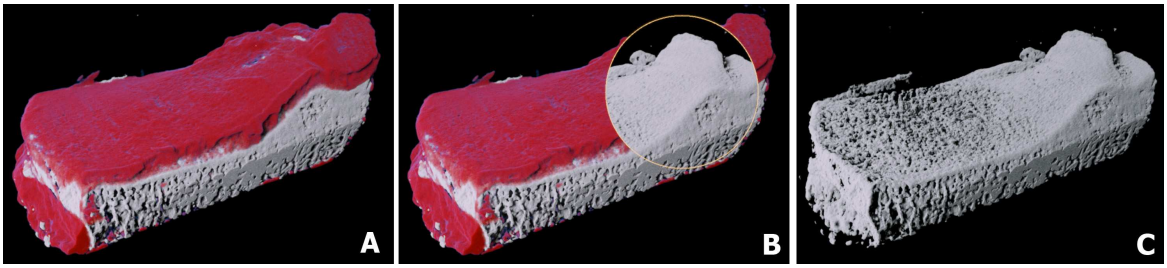


Figure 7.18: Use of the magic lens tool to visualise the Knee Cartilage dataset (section 9.4). **A**: calcium (grey) and iodine (red) **B**: magic lens showing the calcium volume. **C**: calcium volume only.

7.4.2 DVR with a 2D magic lens

The first magic lens type, shown in Figures 7.17 and 7.18, is a circular 2D window fixed to a point on the image canvas. The user positions the centre of the lens by using the mouse cursor, and controls its radius by using a slider. Algorithms 5 and 6 show the modifications made to MARS Vision's DVR algorithm (section 5.3) to support the 2D magic lens. The magic lens algorithm is implemented in CUDA, along with the rest of MARS Vision's DVR code.

Overall, the changes are not significant. The scattering point determination step finds two points: one for the inside of the lens and one for the context. Next, the lighting step uses these two points to determine whether a particular volume should be sampled and shaded.

It must be noted that a circular magic lens shape is not a requirement, and has only been chosen due to the ease of implementation and placement. The criteria for designating pixels as being inside or outside the lens are flexible, and can be changed as needed. For example, a generalised implementation of a 2D magic lens can use an editable binary mask.

Algorithm 5 Changes to MARS Vision's DVR algorithm for choosing a scattering location when the 2D magic lens is enabled. Changes to the original DVR algorithm are shown in red.

Data: Ray R with origin R_o and direction R_d, n volumes $V_{1...n}$, list L of m indices of volumes included in the magic lens ($1 \leq m < n$)

Result: Scattering locations S_{lens} inside the magic lens and $S_{context}$, outside the magic lens

initialization:
lens scattering location $S_{lens} \leftarrow \infty$
context sampling location $S_{context} \leftarrow \infty$
temporary scattering location S_{temp}

if pixel from which R is cast is inside the magic lens **then**
  find the closest scattering point among all volumes that are included in the magic lens:
  for $i \leftarrow 1$ **to** m **do**
    volume index $d = L_d$
    use Woodcock tracking to find scattering point S_{temp} in V_d
    if S_{temp} is closer to R_o than S_{lens} **then**
      $S_{lens} \leftarrow S_{temp}$
    end
  end
else
if the pixel from which R is cast is outside the magic lens, determine closest scattering point using the standard approach:
for $i \leftarrow 1$ **to** n **do**
  use Woodcock tracking to find scattering point S_{temp} in V_i
  if S_{temp} is closer to R_o than $S_{context}$ **then**
    $S_{context} \leftarrow S_{temp}$
  end
end

save S_{lens} and $S_{context}$

Algorithm 6 Changes to the lighting step of MARS Vision's DVR algorithm that are required to support a 2D magic lens. Changes to the original DVR algorithm are shown in red.

Data: Ray R with origin R_o and direction R_d, n volumes $V_{1...n}$, n transfer functions $T_{1...n}$, list L of m indices of volumes included in the magic lens ($1 \leq m < n$), scattering position inside magic lens S_{lens}, context scattering position $S_{context}$

Result: colour C

initialization:
temporary colour TC
temporary opacity TO
total opacity $O \leftarrow 0$

for $i \leftarrow 1$ **to** n **do**
  if pixel is inside the lens **and** $i \in L$ **and** $S_{lens} \neq \infty$ **then**
    if the scattering location inside the magic lens is valid and the current volume is included in the magic lens
    $TO \leftarrow$ calculate opacity at position S_{lens} in V_i using T_i
    $TC \leftarrow$ calculate lighting for position S_{lens} in V_i using T_i
    $TC \leftarrow TC \cdot TO$ *multiply colour by opacity*
    $O \leftarrow O + TO$ *accumulate opacity*
    $C \leftarrow C + TC$ *accumulate colour*
  else if pixel is outside the magic lens **then**
    sample and shade normally
    $TO \leftarrow$ calculate opacity at position $S_{context}$ in V_i using T_i
    $TC \leftarrow$ calculate lighting for position $S_{context}$ in V_i using T_i
    $TC \leftarrow TC \cdot TO$
    $O \leftarrow O + TO$
    $C \leftarrow C + TC$
end

opacity correction
if $O \neq 0$ **then**
  $C \leftarrow C \div O$
end

save C

As Algorithms 5 and 6 show, the 2D magic lens algorithm excludes some volumes from being sampled and shaded inside the lens. This reduces the overall number of calculations and improves the performance of DVR, as shown in Table 7.2. Generally, the increase in performance will vary depending on the number of volumes in the dataset and inside the magic lens, the transfer function and opacity settings, and so on. However, it is clear that the performance of DVR will almost certainly never decrease due to the use of the 2D magic lens.

Table 7.2: Performance of the 2D magic lens in MARS Vision. The same datasets and rendering parameters as those described in section 5.5 are used. The camera is stationary and all optimisations are applied. The performance is measured in iterations per second.

Dataset	No magic lens	2D magic lens	Difference (%)
Mouse12	7.41	10.98	48.19
Spectral Phantom	31.37	35.35	12.70
Meat1127	38.41	49.14	27.93
Mouse0	13.90	16.84	21.17
Knee Cartilage	21.85	22.48	2.88
Plaque108	30.64	38.71	26.23
Plaque72	30.57	32.44	6.13

7.4.2.1 Context preservation mode

Context preservation is an optional mode designed to provide a smoother transition between the inside and the outside of the 2D magic lens. The standard implementation usually produces a sharp discontinuity between the scene inside the lens and the context outside (Figures 7.17 and 7.18). Context preservation causes the context to gradually "fade in" inside the magic lens circle. This effect is shown in Figure 7.19.

The degree of context preservation is controlled by a single parameter, which adjusts the amount of context visible inside the lens. This algorithm is executed if:

(A) the pixel the ray is cast from is inside the magic lens circle, and

(B) the scattering position inside the magic lens (as found by Algorithm 5) is invalid.

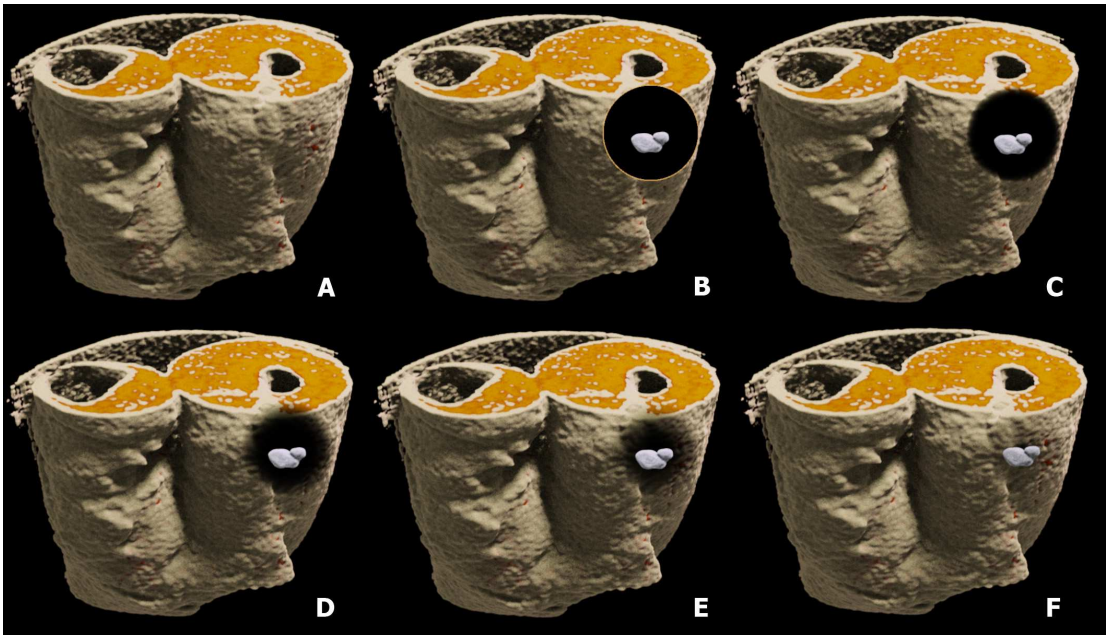


Figure 7.19: Visualisation of the Plaque77 dataset (section 9.2) using a 2D context-preserving magic lens. **A**: No magic lens. **B**: magic lens showing calcium inside the plaque; no context preservation. **C-F**: context preservation turned on. The context preservation parameter is, respectively: 30, 10, 5, 1.

If these conditions are met, then the context preservation algorithm calculates the distance of the pixel from the centre of the lens circle. This distance is expressed as a percentage. The context preservation parameter modifies it according to the following formula:

$$\text{distance} = \text{distance}^{\text{context preservation parameter}} \tag{7.5}$$

where distance ranges from 0 (0%, the centre of the lens) to 1 (100%, the edge of the lens), and the context preservation parameter is an arbitrary value ≥ 1.

Next, the standard rendering algorithm is executed and all volumes are sampled and shaded as usual. The final colour calculated for the pixel is multiplied by the distance from the centre of the magic lens. Therefore, the colour of the pixels at the centre (where the distance is close to 0%) is black, while the pixels close to the edge are virtually unaffected, as shown in Figure 7.19E.

As Figure 7.19 shows, when a high context preservation parameter is used, the result is almost identical to the standard 2D magic lens (B and C). When the parameter is low, then the fade-in effect is more pronounced (D and E). Finally,

if the parameter is 1, then the appearance is identical to the overlay mode (F).

7.4.2.2 Exclusion mode

The default magic lens algorithm implemented in MARS Vision prevents certain volumes from being shown inside the ROI, which is either a moving 2D window, or a 3D sphere. The volumes included in the magic lens are shown both inside, and outside the ROI.

Another possible approach involves only displaying the chosen volumes inside the magic lens, and nowhere else. This is useful for tasks such as visualising the distribution of a certain material inside a particular ROI, and ignoring the distribution of this material elsewhere. In this case, the aim is to remove redundant features from the visualisation shown to the user. This is particularly important for datasets where the selected magic lens volumes occupy a large amount of space, but the ROI is small in comparison.

Exclusion is achieved by changing the lighting step. The exact changes are minor and do not warrant particular interest. In brief, the illumination for all volumes included in the magic lens is not calculated if a pixel is outside it. This, in effect, hides these volumes from the region of the image not covered by the magic lens circle, as shown in Figure 7.20.

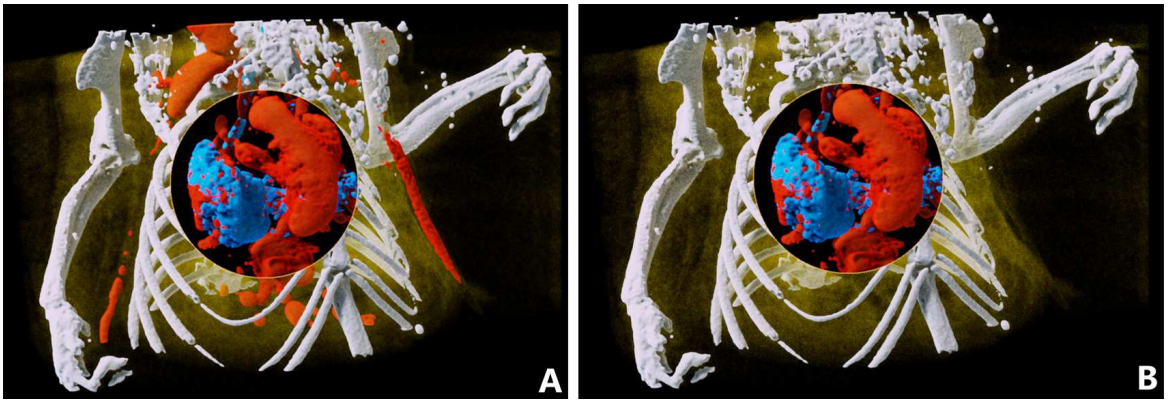


Figure 7.20: Using the 2D magic lens exclusion mode during the visualisation of the Mouse12 dataset (section 9.6.1). **A**: no exclusion. **B**: exclusion mode enabled.

Both magic lens modes serve different purposes. The standard mode retains more context, as it does not hide the chosen volumes outside of the magic lens. Therefore, the transition from the inside to the outside of the lens is smoother. On

the other hand, the exclusion mode presents a much sharper transition between the inside the outside of the lens, but hides irrelevant features from the chosen magic lens volumes.

Therefore, both modes are needed, and each can be used when appropriate. For example, the exclusion mode is useful for visualising the results of some studies into contrast agent detection in a small animal model, described in section 9.6.2. The reason is that the contrast agents used in these studies usually spread across the entire body of the mouse and occlude most internal features during visualisation. However, the ROIs are reasonably small (for example, the liver, the lungs, or the tumours implanted under the skin of the mouse). The exclusion mode allows the user to restrict the visualisation of the contrast agent to a specific ROI, and thus avoid occlusion.

7.4.3 DVR with a 3D magic lens

The 3D magic lens tool is another addition to MARS Vision's DVR algorithm. In this case, the lens is a sphere centred on a point inside the volume, as opposed to the 2D magic lens, which is positioned relative to the image canvas.

This tool is similar to existing clipping tools for cutting out shapes, such as cylinders or spheres, inside the volume, or for cutting out arbitrarily-shaped volumes [295]. However, the 3D magic lens, as implemented in MARS Vision, is designed to be simple to use. Therefore, restrictions have been placed on its shape, limiting it to the spherical shape only.

During visualisation, the 3D magic lens can be used to create a sphere that displays some volumes inside it, while excluding the rest. This means that this type of magic lens can be used to remove occluding tissues inside a chosen ROI. This is the same concept as that behind the 2D magic lens, although the implementation is different.

The user positions the sphere by using the mouse to click on any visible surface in the 3D view. The depth buffer (section 5.4) translates 2D mouse click coordinates to 3D voxel coordinates. If the voxel coordinates are valid, then they are set as the centre of the sphere. The radius of the sphere is controlled by a slider.

The rendering process must only be changed if a ray intersects the magic lens sphere. In that case, Algorithm 7 is executed. Each ray that intersects the sphere finds three scattering points: one inside the sphere, one in front of it, and one

behind. The front and back context locations are found among all volumes, while only the volumes included in the magic lens are sampled to find the lens scattering location. Figure 7.21 shows the ranges where these three points are found.

Algorithm 7 The scattering point selection algorithm executed when a ray intersects the 3D magic lens sphere.

Data: Ray R with origin R_o and direction R_d, n volumes $V_{1...n}$, list L of m indices of volumes included in the magic lens ($1 \leq m < n$), magic lens sphere M

Result: Scattering location S_{lens} inside the magic lens, scattering location S_{front}, in front of the magic lens, and scattering location S_{back} behind the magic lens

initialization:
$S_{lens} \leftarrow \infty$
$S_{front} \leftarrow \infty$
$S_{back} \leftarrow \infty$

$S_{lens} \leftarrow$ determine the closest scattering point inside the magic lens among all volumes in L
$S_{front} \leftarrow$ determine the closest scattering point in front of magic lens among all volumes
$S_{back} \leftarrow$ determine closest scattering point behind the magic lens among all volumes

save S_{lens} and S_{front} and S_{back}

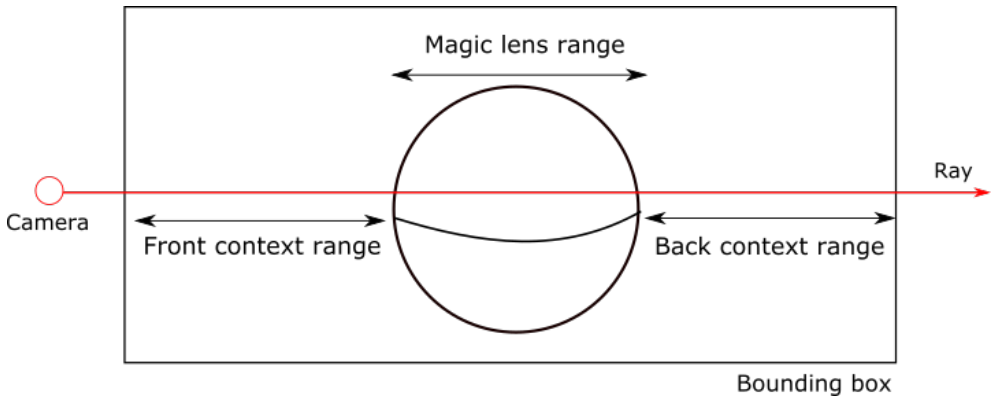


Figure 7.21: Ranges where Algorithm 7 searches for the three scattering locations.

Once the points are found, the lighting step is executed sequentially: first for the front context location, then for the location inside the lens, and finally for the back context location. Invalid locations (that is, the locations where a valid scattering point could not found) are skipped. The result is the appearance of a sphere that excludes certain volumes, as shown in Figure 7.22.

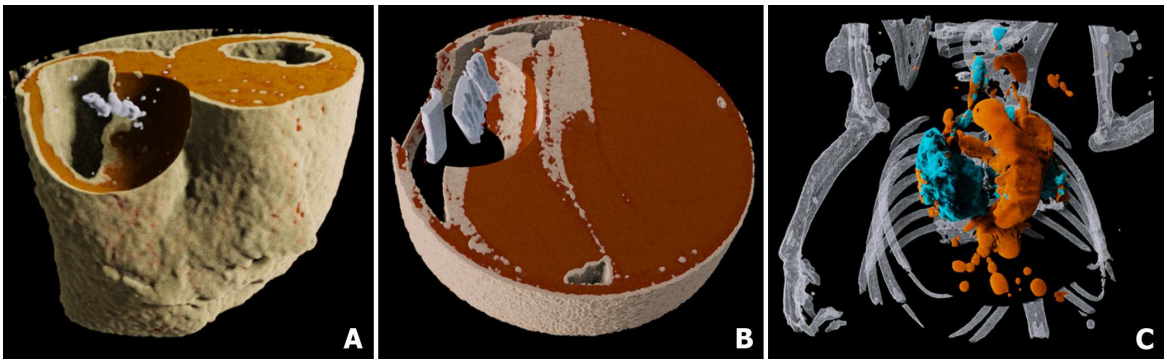


Figure 7.22: Visualisation of three MARS spectral CT datasets using a 3D magic lens. **A**: Plaque77 (section 9.2). **B**: Meat1127 (section 9.3). **C**: Mouse12 (section 9.6.1).

7.4.3.1 Performance

Table 7.3 shows the performance of the 3D magic lens algorithm. For most datasets, the performance is increased. This is interesting, because, in theory, the modifications described in Algorithm 7 should reduce the performance of MARS Vision's DVR algorithm.

In the worst possible case, three separate scattering points (front context, inside the sphere, and back context) will need to be found for a ray, and illumination will need to be calculated at each of these three points. As shown in Table 7.3, this does, in fact, reduce the performance in some cases (for the Knee Cartilage and Plague72 datasets).

In other cases (for the Mouse12, Spectral Phantom, Meat1127, Mouse0, and Plaque108 datasets), the performance is improved because the illumination step performs fewer calculations. As discussed in section 5.3.4, after a sampling point is found, MARS Vision's DVR algorithm conducts one illumination pass for each volume. A 3D magic lens excludes some volumes from a spherical ROI, which means that illumination calculations are *not* performed for these volumes inside the magic lens sphere. This means that, overall, fewer illumination passes are required, and the performance is improved.

Table 7.3: Performance of the 3D magic lens in MARS Vision. The same datasets and rendering parameters as those described in section 5.5 are used. The camera is stationary and all optimisations are applied. The performance is measured in iterations per second.

Dataset	No magic lens	3D magic lens	Difference (%)
Mouse12	7.41	8.95	20.78
Spectral Phantom	31.37	39.39	25.56
Meat1127	38.41	45.68	18.92
Mouse0	13.90	17.15	23.36
Knee Cartilage	21.85	21.76	-0.39
Plaque108	30.64	40.86	33.36
Plaque72	30.57	29.90	-2.19

7.4.4 Slice visualisation with a magic lens

The magic lens tool can also be used during 2D slice visualisation. A moving window can be positioned on a slice, and the contents inside the window can be displayed using different settings.

In MARS Vision, a magic lens is used to display a part of a slice using spectral mode blending (section 5.6.4), while the colour scheme for rest of the image is generated using traditional window and level settings. This technique can be used to retain the anatomical context provided by an energy volume, while displaying material information inside a chosen ROI, as shown in Figure 7.23.

7.4.5 Summary

The magic lens is a versatile tool ideally suited for spectral CT data visualisation. It allows the user to precisely select an ROI and control which volumes are shown inside it. The ROI can be a moving window attached to a 2D image, or a sphere centred on a point in 3D volume space. Further examples of the use of magic lenses for spectral CT data visualisation can be found in Chapter 9.

7.5 Summary

This chapter has described three tools for minimising the effects of occlusion during multi-volume rendering. These tools have been designed to assist with spectral CT data visualisation, but can also be applied to other volumetric data types.

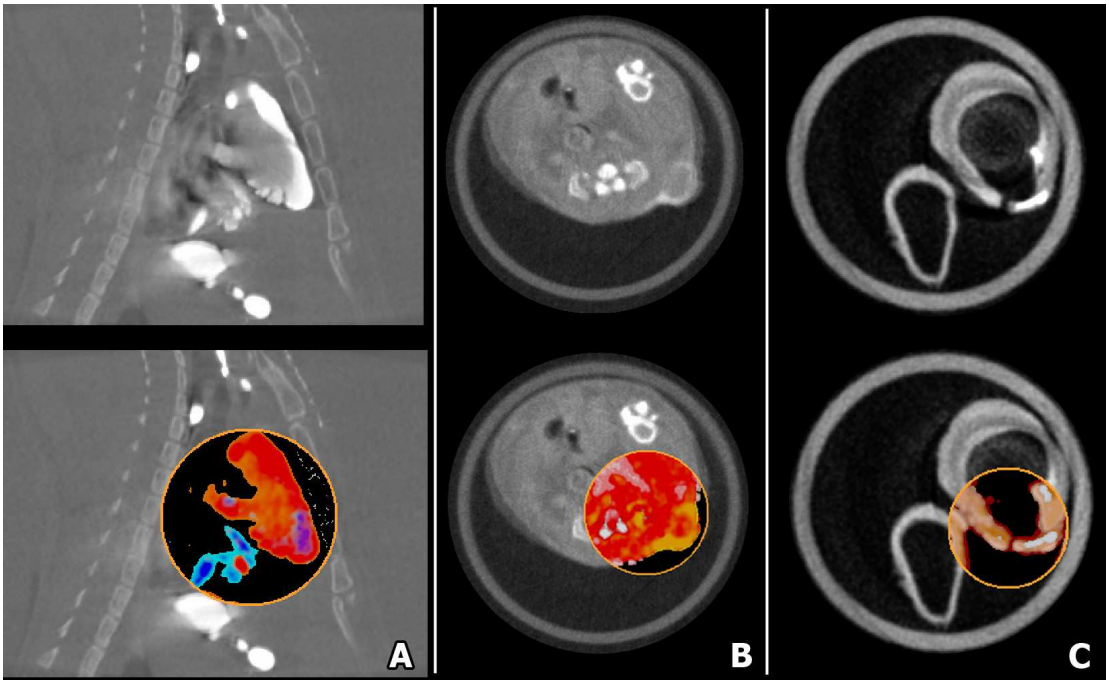


Figure 7.23: Use of the slice view magic lens to visualise three MARS spectral CT datasets. In all cases, multiple material channels are shown inside the magic lens, while the rest of the slice is shown using the greyscale colour scheme generated by window and level settings. **A**: Mouse0 (section 9.6.2).**B**: Mouse12 (section 9.6.1). **C**: Plaque72 (section 9.2).

- Threshold intensity projection is a 3D rendering technique that uses ray marching to find the closest value to a certain threshold along each sampling ray. The colour generated by a ray is dependent of the proximity of this value to the threshold. Threshold intensity projection allows the user to visualise multiple volumes simultaneously and precisely control each volume's visible data range.

- The overlay mode brings a single volume to the front, ensuring that it is always visible. When used during spectral CT data visualisation, this mode is useful for displaying small, occluded ROIs. An example is the visualisation of calcifications in an atherosclerotic plaque.

- The 2D magic lens is a simple interaction technique implemented as a part of MARS Vision's DVR algorithm. It allows the user to move a circular window over the rendered image. Only the volumes selected by the user are

displayed inside the window.

- The context preservation mode is an addition to the 2D magic lens. It provides a smooth transition between the inside, and the outside of the magic lens circle. Starting from the edge of the circle, the context gradually fades to black in its centre, depending on the degree of context preservation chosen by the user.

- A 2D magic lens can be configured to restrict the chosen volumes to being shown only inside the selected ROI. This mode is useful when the distribution of a particular material does not need to be visualised across the entire dataset. This can minimise visual "clutter" and reduce the amount of redundant features present in the scene.

- The 3D magic lens is a sphere in 3D volume space that is centred on a point selected by the user. Only the volumes selected by the user are shown inside the sphere. This provides a more natural-looking alternative to the 2D magic lens.

- The magic lens can also be used during 2D slice visualisation. A moving window can be used to display volumes different to the one currently shown in the slice view widget. This mode is useful for overlaying material information onto energy information, but restricting the overlay to the inside of the magic lens window.

Chapter VIII

Tools for improving visualisation of spectral CT datasets

This chapter describes the auxiliary tools that can be used to assist with the visualisation of current MARS spectral CT datasets. Previous chapters have presented a number of algorithms for rendering these datasets, the GUIs for designing transfer functions, and the techniques for minimising occlusion. This chapter discusses the methods for simplifying the workflow, analysing data, extracting features from energy volumes, and reducing the effects of noise and image artefacts. All tools discussed in this chapter have been implemented in MARS Vision.

This chapter is structured as follows:

- Section 8.1 describes how traditional CT measurement tools can be extended to take advantage of the properties of spectral CT data. It presents a novel tool, called a concentration-volume histogram, which can be used for assessing the spread of a material inside an ROI.

- Section 8.2 discusses the difficulties associated with creating presets for current spectral CT datasets and explains the preset saving and loading functionality of MARS Vision.

- Sections 8.3 and 8.4 explains the function and use of the segmentation and volume data processing tools implemented in MARS Vision. These tools are intended to be used for extracting features from energy volumes.

- Finally, section 8.5 discusses the methods for reducing the effects of noise and image artefacts. This is a serious problem, as various artefacts contained in MARS spectral CT datasets interfere with visualisation by occluding ROIs and distorting the appearance of features.

8.1 Measurements and annotation

Measurements are essential for both medical diagnosis and scientific research. Measurements includes a number of tasks, from measuring the value of a single pixel or voxel, to calculating the amount of a certain material inside a selected ROI. This section explains MARS Vision's measurement tools and describes the concept and implementation of a concentration-volume histogram (CVH) that can be used for analysing the distribution of a material inside an ROI.

MARS Vision groups measurements into two distinct types:

- Measurements of physical size or distance. Examples include the area or volume of an ROI, and the distance between two points.

- Measurements of composition. Examples include measuring the value of a single pixel, and the average, median, standard deviation or variance of pixels or voxels inside an ROI.

Analysis of conventional single-energy CT datasets includes measurements of both size and composition of ROIs [81,238]. However, the composition information is usually limited, because a single energy range measured by conventional CT (usually, the human diagnostic energy range of 10-120 keV [24]) does not provide sufficient information to quantitatively discriminate between materials [157].

Spectral CT provides anatomical information through energy volumes and molecular information through material volumes. This means that measurement becomes more important, as precise measurements of the composition of ROIs become available. Therefore, conventional measurement tools must now be extended to take advantage of the properties of spectral CT data.

This section describes the implementation of both types of measurement tools. Previously, the MARS group performed analysis and measurement of spectral CT datasets using general-purpose image processing tools. This is because no custom measurement tools for spectral CT data appear to have been designed (based on the survey of available literature on spectral CT).

This section describes two tools designed to enhance the presentation of results of spectral CT data analysis. The first (section 8.1.2) is an extension of the well-known profile of a line graph to simultaneously display the profiles of multiple energy or material volumes. The second, the concentration-volume histogram

(CVH), is a novel way of measuring the distribution of a material inside a chosen ROI (section 8.1.3).

8.1.1 Distance, area and volume measurements

As mentioned above, size, area, volume and distance measurements are applicable to all variants of x-ray computed tomography. MARS Vision provides this functionality to the user by offering the choice of three measurement tools and extends it to allow the user to simultaneously perform measurements of multiple volumes, as shown in Figure 8.1.

Individual pixels can be selected by using the mouse cursor in the 2D view (white arrow in Figure 8.1A). The values of all pixels at the selected location are displayed by a dedicated GUI element, as shown in Figure 8.1B. In addition, lines can be measured, and outlines of polygons can be drawn (Figure 8.1C). Lines and polygon ROIs can be assigned a name, and can therefore also serve as annotations.

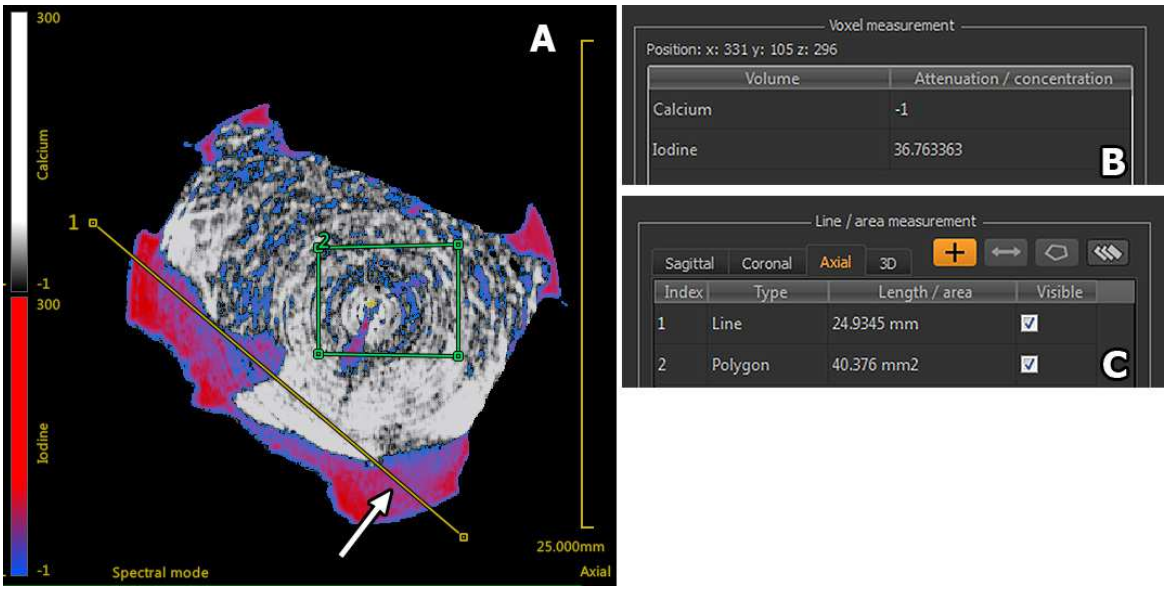


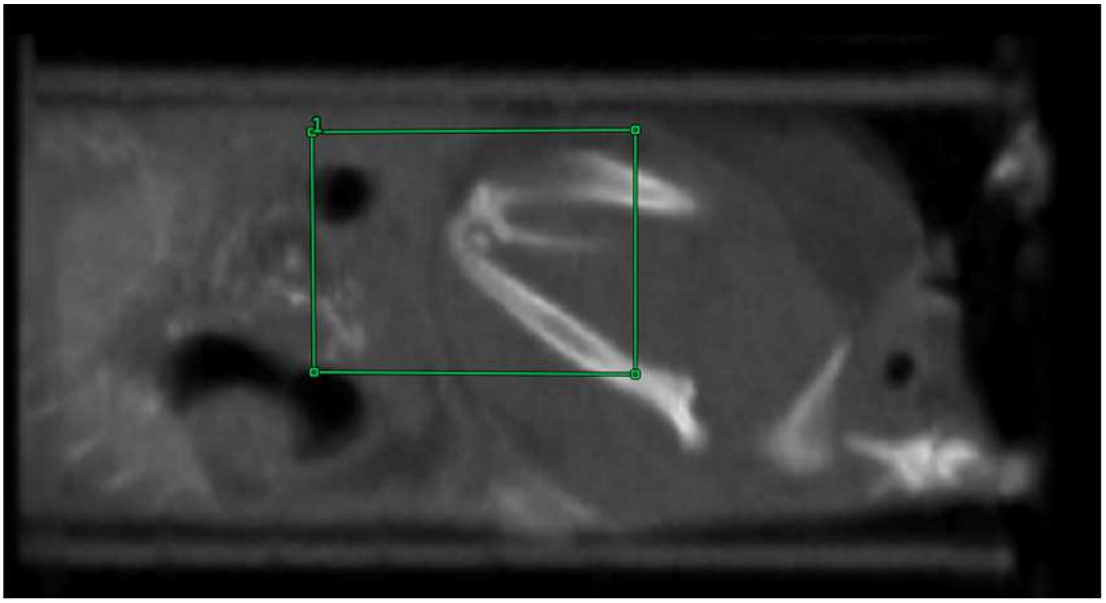
Figure 8.1: Measurement of the Knee Cartilage dataset (section 9.4). **A**: a 2D spectral mode visualisation of the calcium and the iodine channels. The colour gradients are shown on the left. A line, a polygon, and a single pixel measurement (marked by an arrow) are shown. **B**: the GUI for showing the value of a single pixel in multiple volumes. **C**: the GUI for showing the list of line and polygon ROI measurements.

8.1.2 Adapting volume composition and statistics tools to spectral CT

Measurements of the composition of ROIs is an important task in spectral CT data analysis. Such measurements are useful for both energy [9] and material [10] volumes of a spectral CT dataset. If energy volumes are measured, then the average data value inside an ROI can be compared across different volumes. If material volumes are measured, then the amount of a material inside an ROI can be calculated.

The polygon drawing tools described above can also be used to generate a set of pixels contained within the polygon ROI. This allows us to calculate basic statistics, such as the mean, median, mode, minimum and maximum data values, the standard deviation and the number of pixels. The statistics of all volumes inside the selected ROI can be measured simultaneously, as shown in Figure 8.2.

The standard calculation of the profile of a line (also referred to as "line profile" for brevity) involves measuring the pixel values along a line and plotting them on



Volume	Slice	Type	Min	Max	Average	Median	Mode	Std. dev.	No. voxels
8.0 keV - 27.9 keV	230	Polygon	-0.000413693	0.873095	0.353208	0.315154	0.309691	0.126976	15195
27.9 keV - 48.2 keV	230	Polygon	-0.0145184	0.73096	0.298512	0.271453	0.265991	0.102936	15195
48.2 keV - 68.1 keV	230	Polygon	-0.0148385	0.483691	0.216753	0.205903	0.205903	0.0598805	15195
68.1 keV - 81.2 keV	230	Polygon	-0.0155426	0.41417	0.197945	0.189515	0.189515	0.0485142	15195

Figure 8.2: Simultaneous measurement of slices from four energy volumes of the Mouse0 dataset (section 9.6.2) using MARS Vision's ROI statistics tool.

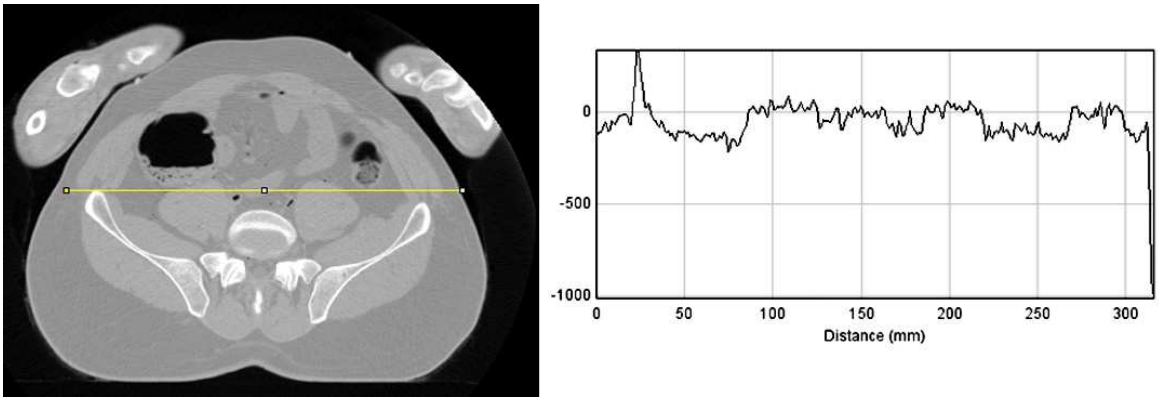


Figure 8.3: Using ImageJ to measure the profile of a line drawn across a slice of the Visible Human Dataset (section 9.7.1).

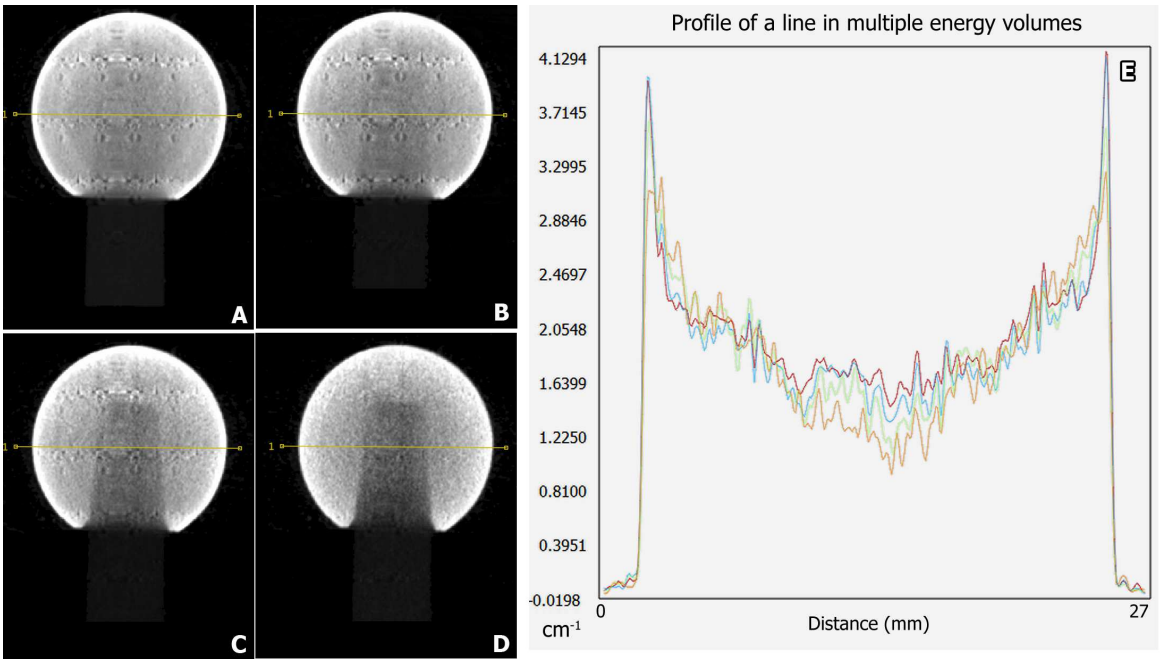


Figure 8.4: Using MARS Vision to measure the profile of a line drawn across a slice of the CoCr Ball dataset (section 9.5.2). **A**-**D**: the 50-120, 60-120, 70-120 and 80-120 keV energy volumes, respectively. **E**: the line profile graph, showing all four lines simultaneously. The colours are, respectively, red, blue, green and yellow. Note that the "dip" in the middle of the profile graph corresponds the hollow space inside the ball-shaped implant.

a graph. The graph shows the variation in the pixel values over a certain distance. This information can be used to assess the spread of a material or the severity of beam hardening.

Extending this tool to spectral CT data analysis involves plotting multiple lines on the same graph (Figure 8.4). This allows users to compare the attenuation profile across all energy volumes, or to plot the variation in the concentrations of multiple materials on a single graph.

8.1.3 Concentration-volume histogram

This section discusses the concept and implementation of a concentration-volume histogram (CVH), a tool for evaluating the distribution of a material inside a chosen ROI. It is similar to the dose-volume histogram (DVH), which is commonly used for treatment planning in radiotherapy [296]. A DVH can be used to graphically present the radiation dose received by different organs, as shown in Figure 8.5. This allows the physician to compare possible treatment plans, and select the one that gives the smallest possible radiation dose to the patient, or to a specific organ. Further, a DVH shows the percentage of an organ that is receiving a certain dose, which is valuable for assessing the damage that radiotherapy can do to healthy tissue.

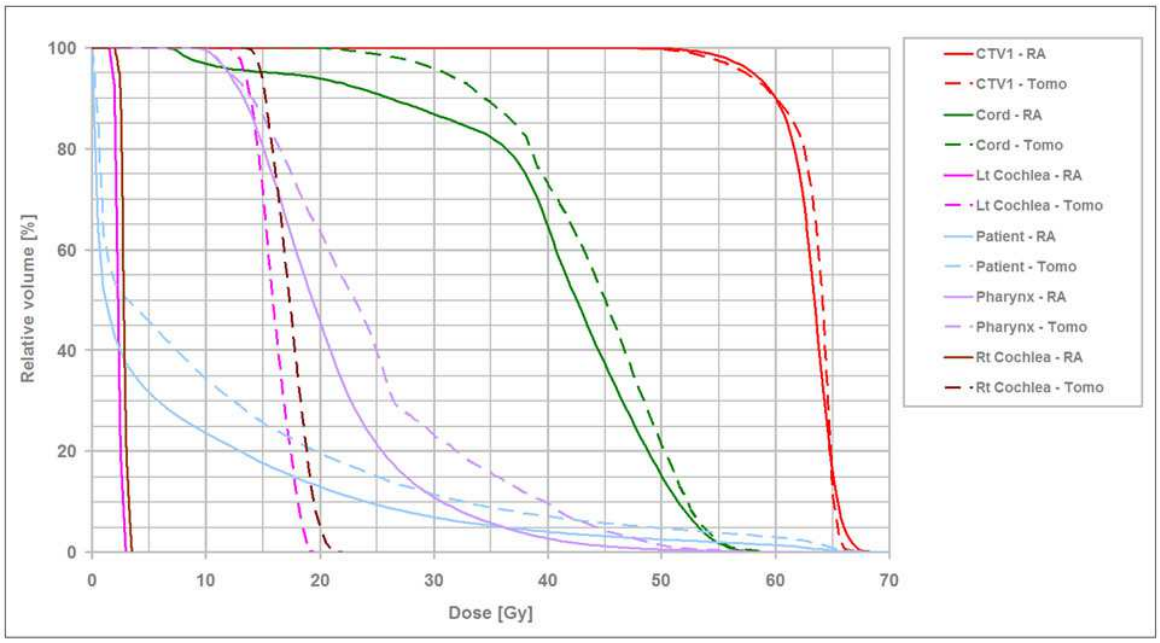


Figure 8.5: A conventional cumulative dose-volume histogram used in radiotherapy. The x-axis shows the radiation dose received by an organ, while the y-axis shows the percentage of an organ receiving that dose. The lines show the doses received by different organs.

For spectral CT imaging, the concept of a DVH can be replaced by the concept of a CVH. This tool has been created after consulting with the pre-clinical researchers from the MARS team, who were familiar with DVHs and wished to use a similar visualisation technique for spectral CT data. Like a DVH, a CVH shows a plot of the percentage of a region or an organ receiving a certain dose. However, the radiation dose is replaced by the concentration of a material.

Therefore, a CVH is a way of presenting the molecular information provided by spectral CT. The concentration is plotted along the x-axis, and ranges from the minimum concentration of a material inside the ROI to the maximum concentration. The y-axis is unchanged. An example of a CVH is shown in Figure 8.6D.

This section only discusses the creation of a CVH for a 2D ROI, as MARS Vision does not currently support 3D ROI measurement (due to the difficulty of precisely selecting ROIs during DVR visualisation). However, the CVH is a generic concept that can be applied to an ROI of any shape.

In MARS Vision, existing tools for drawing a 2D ROI can be used to mark an area in one of the slice views (Figure 8.6C). Once the ROI has been designated, all pixels inside it are retrieved and the CVH generation algorithm is executed. This algorithm is described below:

1. The data range of the measured ROI is partitioned into an arbitrary number of bins. For example, if the concentration range is 0-20 mg/ml and the number of bins is 100, then the first bin will count all pixels between 0 and 0.2 mg/ml and the last bin will count all pixels between 19.8 and 20.0 mg/ml.

2. Each pixel inside the ROI is compared to the minimum threshold of each bin. If the pixel value is equal to or below the threshold, the number of pixels in the bin is incremented by one.

   Therefore, the first bin will always contain all pixels in the ROI; the second bin will only contain the pixels above a certain concentration threshold, and so on. Following the example above, the first bin (0-0.2 mg/ml) will contain all pixels in the ROI, while second bin (0.2-0.4 mg/ml) will likely contain fewer pixels.

3. After all pixels have been evaluated, the number of pixels in each bin is

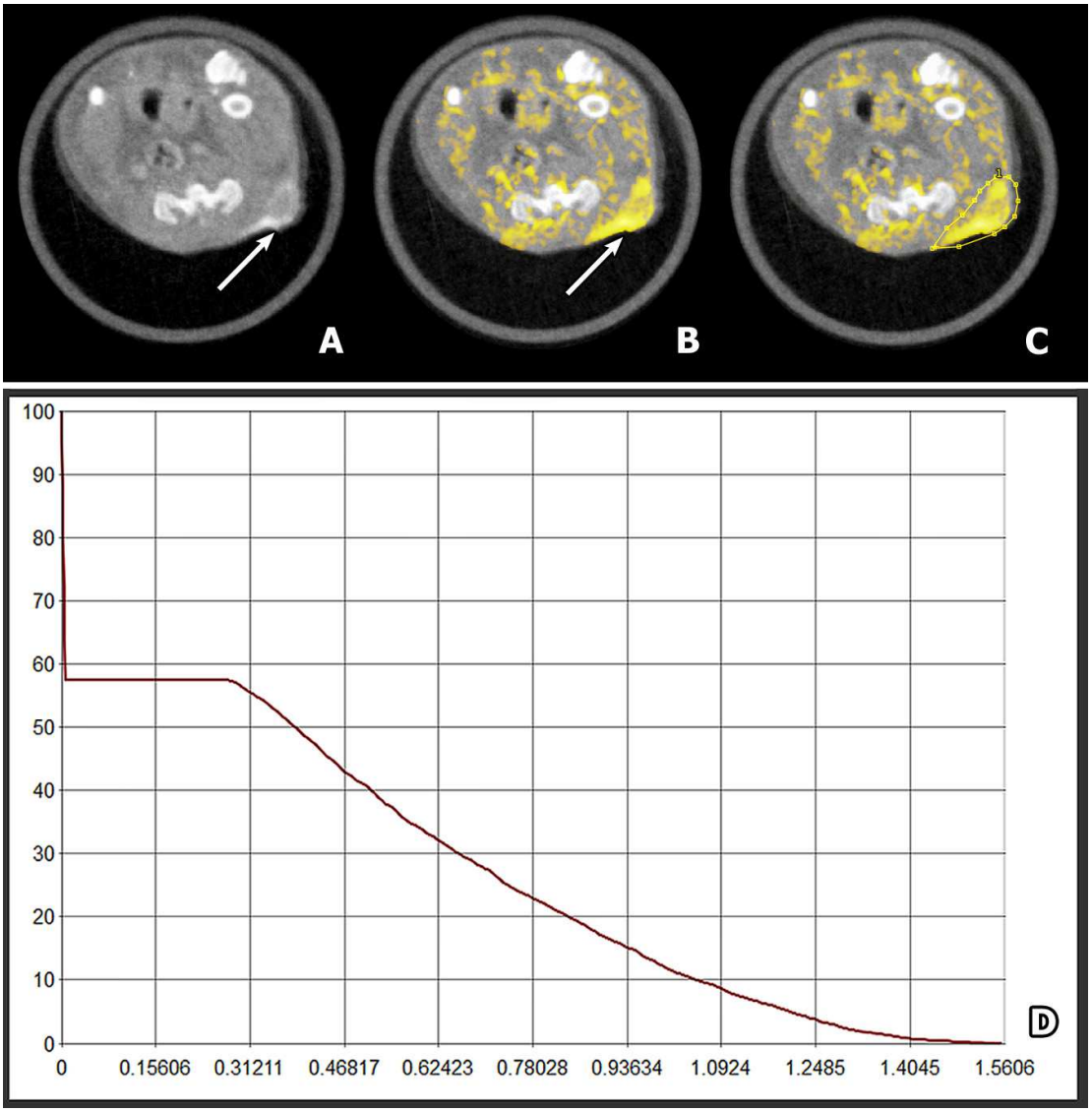


Figure 8.6: Using a concentration-volume histogram to evaluate the distribution of gold inside the Mouse0 dataset (section 9.6.2). **A**, **B**: Identifying the ROI using an energy volume, or an energy volume with a material overlay. **C**: Drawing a polygon to enclose the ROI. **D**: The resulting CVH. The x-axis shows the concentration of a material (gold, in mg/ml) inside the ROI. The y-axis shows the percentage of pixels in the ROI that are at least this concentration.

converted to a percentage by dividing by the total number of pixels in the ROI.

The results are presented as a graph, as shown in Figure 8.6D. MARS Vision combines visualisation and measurement, so the 2D visualisation and data fusion

tools described in section 5.6 can be used to examine an energy volume, a material volume, or both at once. Once the ROI is found, the distribution of a material inside it can be measured, and the information can be presented using a CVH.

Figure 8.6 shows one current use case for a CVH. The mouse used in this study has been injected with gold nanoparticles (AuNP), which have accumulated inside the tumour implanted under its skin (highlighted by an arrow). Researchers investigating the use of AuNP as a spectral CT contrast agent can measure its distribution inside the tumour using a CVH. This information can be used to calculate how much gold reaches the tumour, and which percentage of the tumour has accumulated a certain concentration of gold.

In this study, gold is only used to highlight the location of the ROI (not as a therapeutic agent), but the same approach can also be used during the drug discovery process. For example, if a mouse is injected with a drug targeted at a particular type of cancer, then a CVH can be used to measure whether a sufficient amount of this drug has reached the tumour, and which concentration appears to be effective at treating it. Further, the same animal may be scanned multiple times, and CVHs created after each scan may be compared to monitor the effects of the drug over time.

In the future, a CVH may be used in clinical practice for evaluating the effectiveness of treatment. The process is the same: a drug (perhaps one specifically formulated to be detectable by spectral CT) may be given to the patient, who is subsequently scanned by a spectral CT system. Material decomposition identifies and quantifies the drug. Finally, a CVH is used to determine whether it has reached its target (such as a tumour), and which percentage of the target has received a sufficient concentration. Comparing CVHs created after multiple scans may help to determine whether a condition is responding to treatment, and whether the drug dosage needs to be increased.

It must be noted that a CVH is not the only tool that can be used for visualising the distribution of a material inside an ROI. As stated above, it can be applied to a 2D or 3D ROI (or even multiple ROIs) of any size and shape, but it can also be used in combination with other visualisation techniques. For example, a CVH can be displayed together with a height field [297]. In this case, the CVH will provide quantitative information, while the height field will show the user where regions of a particular concentration of a material are located. Combining different techniques in this manner is a well-known principle of data visualisation [298], which has been found to be particularly useful for multi-variate data visualisation [120].

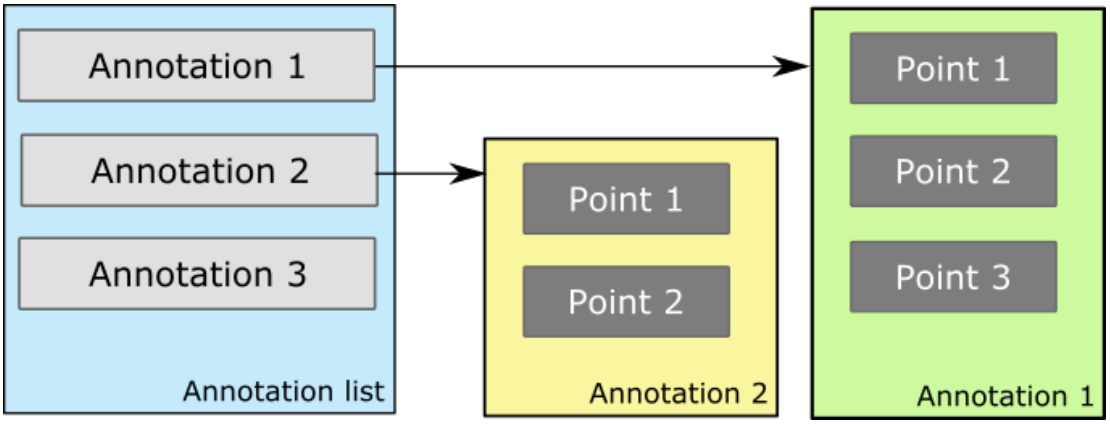


Figure 8.7: MARS Vision's 3D annotations list. Each annotation in the list consists of a set of points, along with a name, opacity, colour and a reference to the volume the annotation is attached to.

8.1.4 3D annotations

The placement and manipulation of annotations are an important part of a radiologist's workflow because annotations can be used to mark ROIs and communicate this information to colleagues. Alternatively, annotated datasets can be used for education [299, 300].

MARS Vision's DVR visualisation (section 5.3) supports 3D point annotation placement. Similar to the 3D magic lens (section 7.4.3) and the voxel measurement described above, the mouse is used as a selection tool to place points on visible surfaces. The depth buffer is used to map 2D mouse click coordinates to coordinates in 3D volume space.

Figure 8.7 shows the structure of MARS Vision's 3D annotations list. Each 3D annotation consists of a list of points. The user works with one annotation at a time by choosing which annotation to edit. All points placed in the 3D view will be added to that annotation's point list.

Annotation points are stored in an XML file, along with other visualisation parameters (such as transfer function settings and camera position), as described in section 8.2. This limits their utility, as annotations made to the same dataset on different workstations cannot be synchronised. An example of XML code for storing annotation data is provided in Appendix B.

In the future, 3D volume annotations, together with 2D slice annotations and measurements, will be integrated into the overall DICOM-based toolchain. Anno-

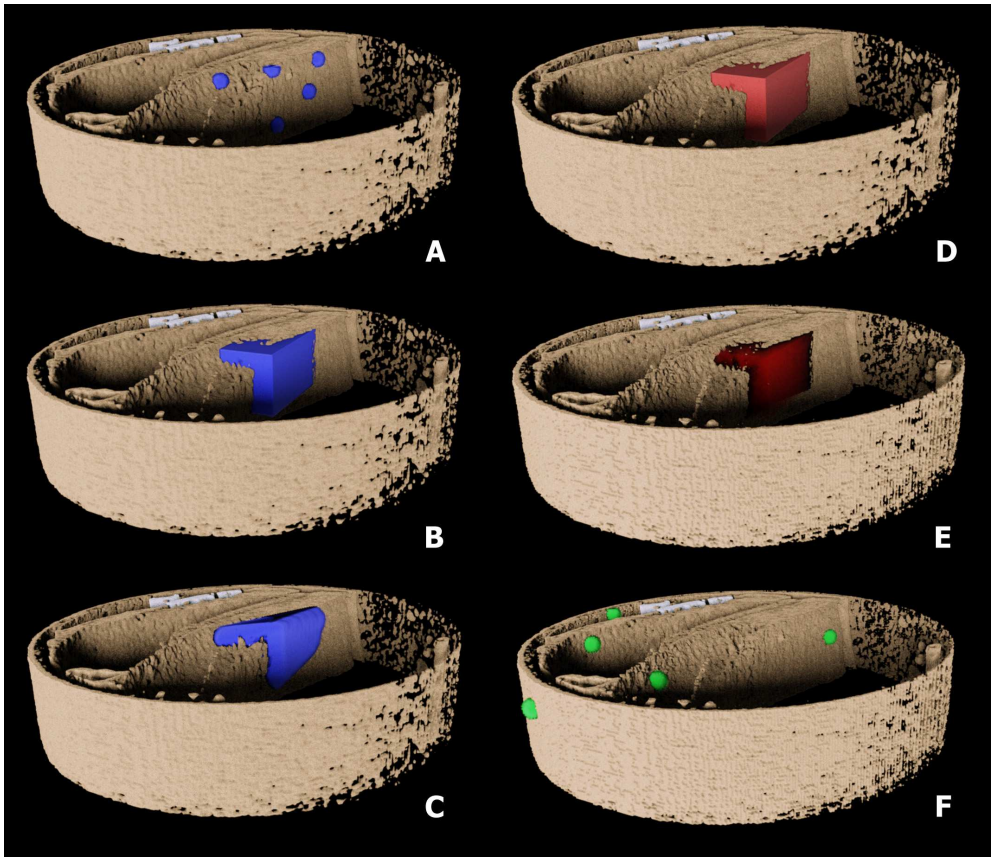


Figure 8.8: Annotating the Meat1127 dataset (section 9.3.1). **A**: Points. **B**: Box. **C**: Convex hull. **D**: Same annotation as **B**, but different colour. **E**: Same annotation as **B**, but lower opacity settings. **F**: Different annotation only consisting of points.

tation storage will likely be based on the DICOM annotation specifications (DICOM Standard Part 3, Annex A [301]). These specifications provide support for "overlay images", which allow annotations to be saved as a separate image without changing the original slice data.

Point annotations can be extended to mark out a volume of space. This is done by creating a bounding box that encloses all points in a particular annotation (Figure 8.8B), or the convex hull (Figure 8.8C), which creates the smallest possible 3D shape that contains all points [302]. Once the convex hull is created, it is dilated and blurred using 3D Gaussian blur to smooth out its shape and reduce block artefacts.

The annotation, consisting of points, a bounding box, or a convex hull is painted into a volume and rendered like all other energy and material volumes. Colour

(Figure 8.8D) and opacity (Figure 8.8E) can be adjusted. Only one annotation may be displayed at a time, but the visible annotation can be changed at any time (Figure 8.8F). Finally, annotations can be "attached" to a particular volume. This means that the annotation will only be rendered if the parent volume is visible.

Annotations may obscure small features of interest, so the user is always given a choice to toggle annotation rendering on and off. Finally, a text string is associated with each annotation to enable users to describe features of interest and communicate this information to other MARS team members.

8.1.5 Summary

This section has explained the implementation of basic measurement tools for analysing the size and the composition of ROIs in MARS spectral CT datasets. In conclusion:

- The tools for simultaneous measurement of multiple energy and/or material volumes have been developed by extending traditional profile of a line and ROI statistics measurement tools.

- The concentration-volume histogram, which is a novel tool for assessing the distribution of a material inside an ROI, has been described and implemented.

- ROIs inside MARS spectral CT datasets can be labelled by using the 3D annotations tool. This is done by using the mouse to place points on visible surfaces during DVR visualisation. These annotations are saved in presets files (described in the next section) that can be shared between team members.

8.2 Presets

Presets can be used to simplify the user's workflow by providing certain default settings that emphasise particular features of interest and hide other features. Since the preferences and workflows of users vary, presets are largely subjective, and are most commonly found by experimentation [81]. Support for saving and loading user-defined presets is a standard feature of DICOM viewers [79, 81, 238].

In clinical CT imaging, different presets are used for displaying the structure inside bones, for showing the variation in the radiodensity of soft tissue, for highlighting the distribution of a contrast agent within a patient's body, and so on.

Figure 8.9 provides an example of presets used for 2D slice visualisation of conventional CT datasets. Similarly, transfer function presets can also be created for 3D rendering (for an example, refer to Figure 6.3). However, these presets can only exist if the data formats are standardised. In clinical practice, the Hounsfield scale is used to represent the data inside energy volumes (section 3.2.1.1), but there is no accepted standard for spectral CT data.

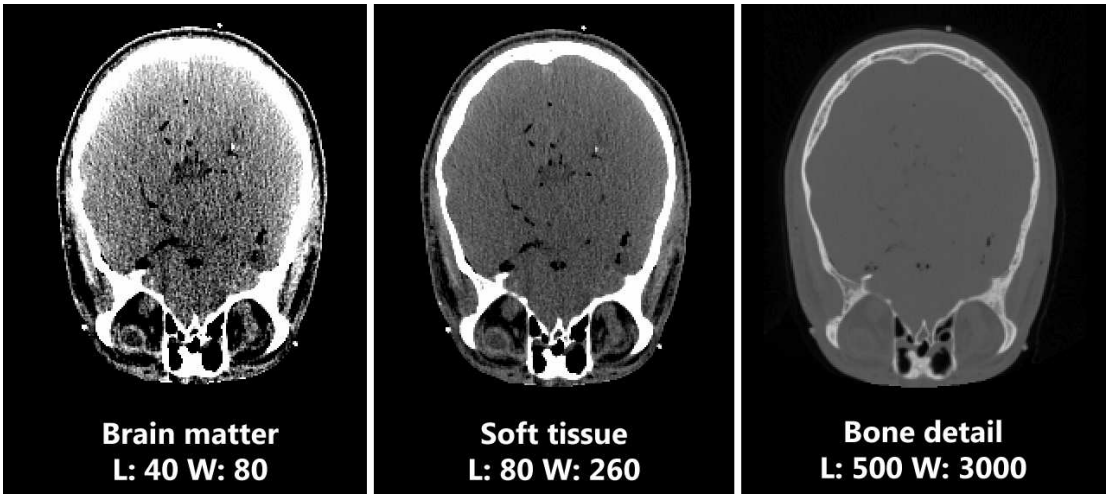


Figure 8.9: Visualisation of a slice of the Visible Human Male dataset using three different presets. The level and window values are given in Hounsfield Units.

Therefore, generic presets that apply to all spectral CT datasets cannot yet be created. Support for such presets will likely be added to MARS Vision in the future, once the data formats have been standardised. Currently, MARS Vision only supports dataset-specific presets.

8.2.1 Dataset-specific presets

Users working with MARS spectral CT datasets usually explore and visualise a dataset multiple times. Therefore, saving a collection of settings that describes the appearance of a dataset is a good way of accelerating the user's workflow. This collection is referred to as a dataset-specific, or a "local" preset. Local presets are only applicable to the datasets they have originally been created for, and cannot

be shared across datasets.

Local presets save all the information required to restore MARS Vision to a certain state. This includes camera settings, lighting parameters, transfer function settings and measurements and annotations.

View settings describe the size of the canvas and the orientation of the camera. Light settings describe the position, intensity and colour of all lights in the 3D DVR scene. Transfer function settings describe the position, colour and opacity of all nodes in each transfer function graph. Measurements are the 2D lines and polygon ROIs described in the previous section and annotations are the points placed by the user in the 3D view (section 8.1.4).

Saving a local preset generates an XML file that stores these settings, together with a small thumbnail image to illustrate the dataset's appearance when this preset is applied. An example of a presets file is provided in Appendix B.

The GUI for accessing local presets is shown in Figure 8.10. The preset is loaded when the user clicks on a thumbnail.

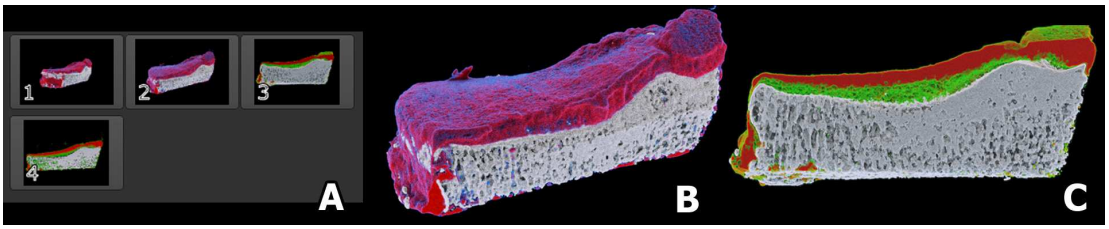


Figure 8.10: Examples of local presets for the Knee Cartilage dataset (section 9.4). **A**: Local preset manager GUI showing 4 different presets with thumbnails. **B**: appearance of the dataset when preset 2 is applied. **C**: appearance of the dataset when preset 3 is applied.

8.2.2 Summary

MARS Vision implements basic preset saving and loading functionality. All parameters needed to restore the state of 3D DVR visualisation and all 2D line and polygon measurements are saved. Currently, all presets are specific to a particular dataset and cannot be applied to other datasets.

8.3 Segmentation

This section describes the implementation of threshold-based segmentation tools in MARS Vision. This thesis is not concerned with segmentation of spectral CT datasets (that is, investigating how the unique properties of spectral CT data can be exploited to improve segmentation), which is a separate research topic. Instead, basic single-volume segmentation has been implemented, as it is a useful addition to the overall toolset, especially when combined with the magic lens or overlay tools (described in Chapter 7).

In the context of CT imaging, segmentation refers to the partitioning of a volumetric dataset according to certain rules in order to extract features. Usually, this process creates a new volumetric dataset or 2D slice. This is different from the classification that can be achieved by creating a transfer function for DVR visualisation (as described in Chapter 6). Transfer functions achieve purely visual separation of materials or features, while segmentation typically extracts all data that fits a particular set of criteria into a new dataset.

In clinical practice, segmentation can be used to separate the organs in the abdomen [272], analyse the composition of bones [303], extract the lungs [273] and analyse liver structure [304].

In the MARS software toolchain, a form of segmentation is performed by material decomposition (section 3.2.2), which extracts material information from reconstructed energy volumes. However, energy volumes can also be segmented during visualisation. This allows users to preview the results and to adjust the segmentation parameters, if necessary.

MARS Vision allows the user to initiate segmentation from any of the 2D slice views, while the results of segmentation can be displayed in both 2D and 3D. A single slice may be segmented, as shown Figure 8.11. Alternatively, segmentation can be applied to an entire volume, as shown in Figures 8.12 and 8.13.

Four segmentation types are available. In all cases, the user controls the segmentation parameters by adjusting the upper and lower thresholds.

1. Region growing segmentation in a slice. This type extracts all pixels between the lower and the upper thresholds from a given slice. Segmentation is seeded by selecting a starting point in one of the 2D slice views, or by clicking on a visible surface in the 3D view. In this case the depth buffer (section 5.4) is

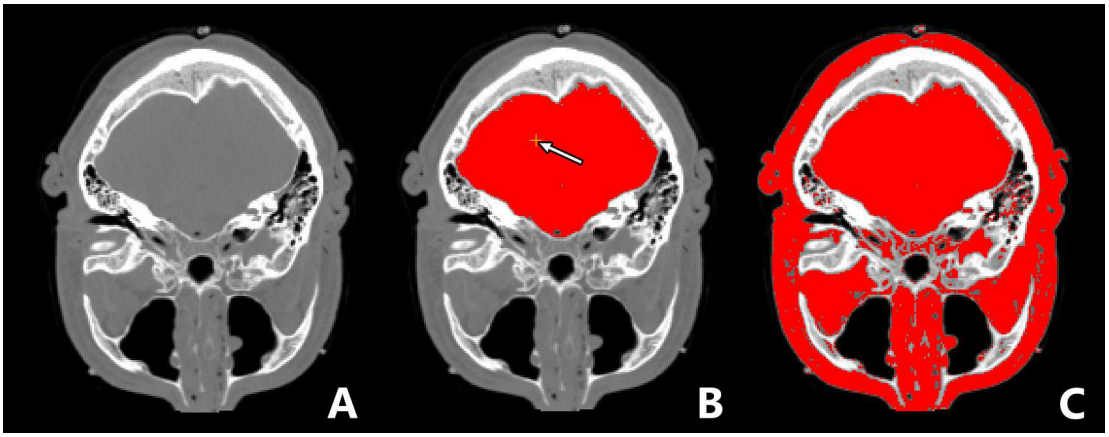


Figure 8.11: Segmenting a slice of the Visible Human Male dataset (section 9.7.1). **A**: no segmentation. **B**: region-growing segmentation initiated from a point marked by an arrow. **C**: whole-slice segmentation. The lower threshold is at -60 HU and the upper threshold is at 120 HU.

used to map the mouse click coordinates to the 3D voxel coordinates. The result is shown in Figure 8.11B.

2. Whole slice segmentation, which extracts all pixels between the lower and upper thresholds from a given slice, as shown in Figure 8.11C.

3. Region growing segmentation in a volume. This algorithm grows a volume from a selected seed point. All segmented voxels are placed into a new volumetric dataset, as shown in Figure 8.12.

   Region growing segmentation is an iterative process that starts at the selected voxel coordinate and moves in both directions along the x, y, and z axes. The voxels that are between the upper and the lower thresholds are added to the segmented volume.

   New segmentation passes are launched from the voxels that have just been added to the segmented volume. Iterations continue until no more valid voxels can be reached.

4. Whole volume segmentation. This type extracts all voxels between the lower and the upper thresholds from a given volume and creates a new volumetric dataset. The segmentation algorithm involves sequentially applying whole

slice segmentation to all slices of the given volume. The results are shown in Figure 8.13

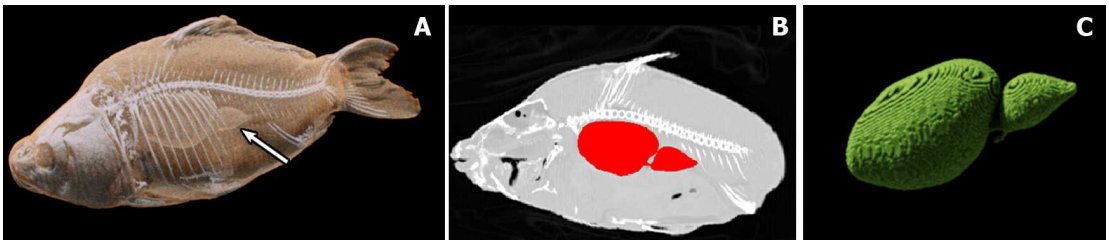


Figure 8.12: Extraction of the swim bladder from the Carp dataset using region growing segmentation. **A**: no segmentation. The swim bladder is marked by an arrow. **B**: swim bladder segmented out in a single slice. **C**: segmented volume visualised using DVR.

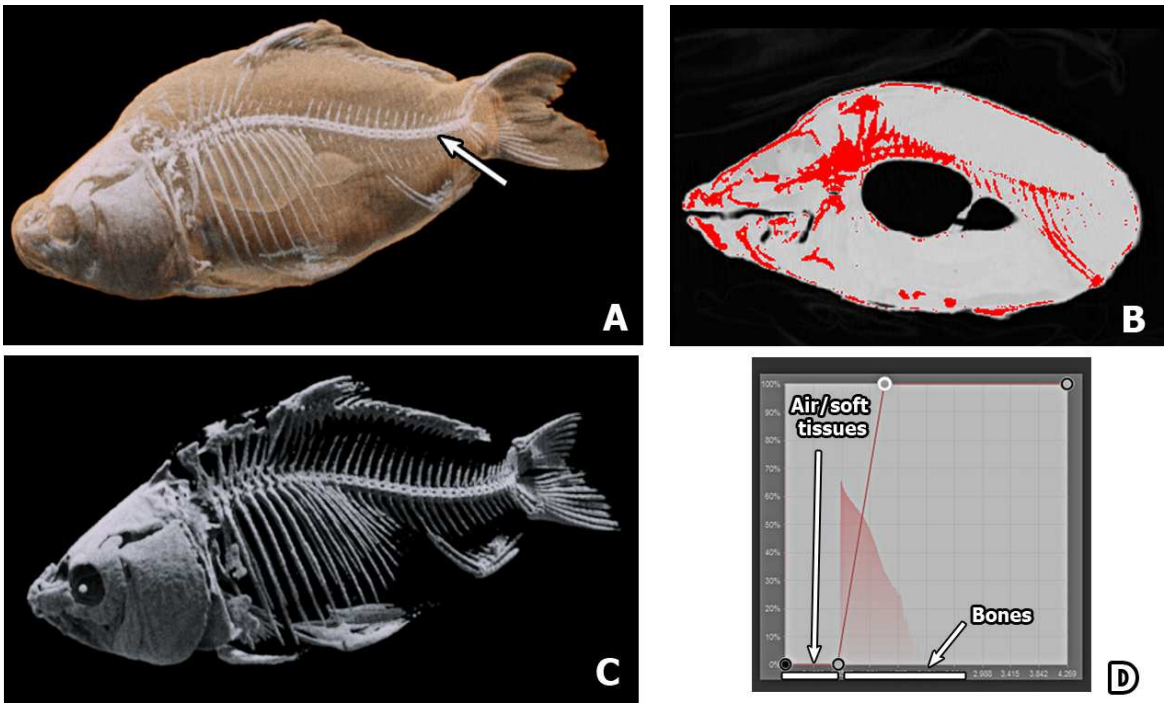


Figure 8.13: Extraction of bones from the Carp dataset using whole volume segmentation. **A**: no segmentation. The bones are marked by an arrow. **B**: bones segmented out in a single slice. **C**: segmented volume visualised using DVR. **D**: the transfer function used in **C**. Note that the histogram is missing a region corresponding to air and soft tissue.

In addition, all slice segmentation can be restricted to the inside of polygon drawn by the user (using the tools described in section 8.1.1), and pixels can be added to or removed from the selection by directly painting with the mouse.

8.3.1 Use of segmentation

The segmentation tools described above are not specific to spectral CT. Nevertheless, they can be used to improve the visualisation of spectral CT datasets. In particular, segmentation can first be used to extract features from an energy volume into a separate volumetric dataset. Next, a magic lens or an overlay can be used to visualise these features without occlusion, while the original volume can be used as the context. This method is shown in Figures 8.14 and 8.15.

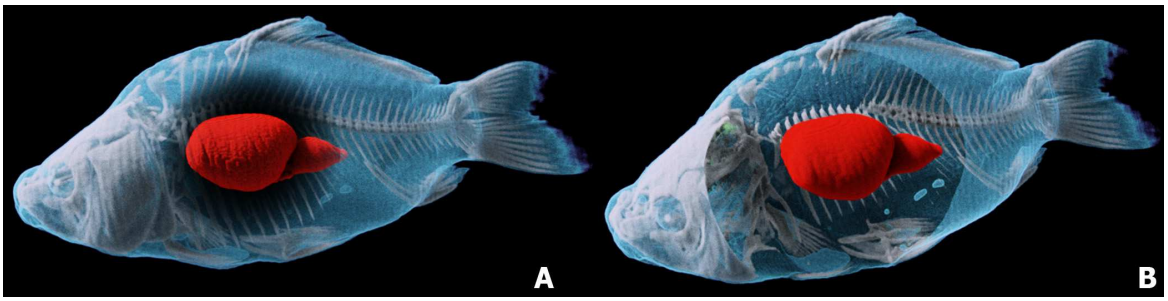


Figure 8.14: Use of the magic lens to visualise the swim bladder extracted in Figure 8.12. **A**: 2D context preserving magic lens. **B**. 3D magic lens centered on the swim bladder.

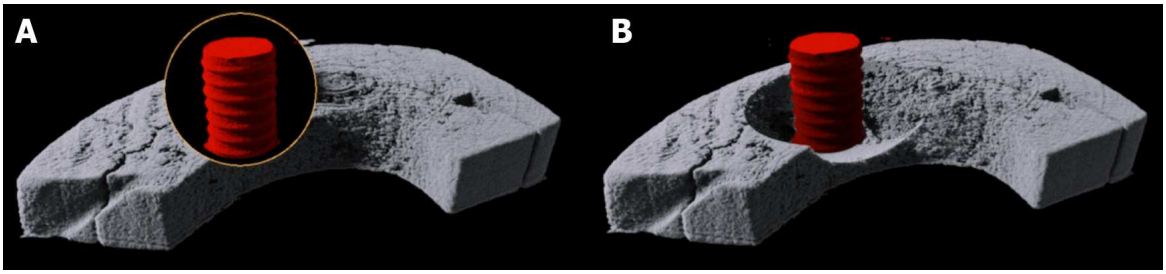


Figure 8.15: Use of the magic lens to visualise the screw extracted from an energy volume of the TiScrew dataset (section 9.5) using whole volume segmentation. **A**: 2D non-context preserving magic lens. **B**. 3D magic lens centred on the screw.

8.3.2 Limitations due to noise

Threshold-based segmentation is sensitive to noise, which is often present in MARS datasets, as described in section 3.3. Therefore, improved implementations of segmentation algorithms for spectral CT data should employ more advanced techniques, such as adaptively changing the segmentation threshold based on the properties of the segmented region and the neighbouring regions [305, 306], adding spatial or geometric constraints [307], or incorporating prior knowledge about the structures or organs being segmented [306, 308].

8.4 Volume data processing

This section describes the interactive volume data processing tools implemented in MARS Vision. The processing discussed in this section refers to the extraction of materials or other features of interest through the use of various arithmetic operations.

Earlier research by the MARS team has shown that material information can be extracted from spectral CT energy volumes by arithmetic operations such as the addition or subtraction of two or more volumes [129]. However, currently, the majority of data processing is done prior to visualisation (section 3.1) and material extraction by interactive intermixing of energy volumes is no longer conducted (section 4.3.2.1). Instead, it has been superseded by the visualisation of material volumes.

8.4.1 Use cases

First, operations, such as clamping the values of all voxels in a volume to a certain minimum or maximum value allow for primitive segmentation or noise removal.

Second, as explained in section 4.3, side-by-side comparison of slices of different energy volumes is a popular spectral CT data visualisation technique. Slices are usually shown next to each other, but a subtracted image may also be provided to clearly visualise the difference between two bins, as shown in Figure 8.16. Alternatively, non-overlapping energy volumes (for example, 50-80 and 80-120 keV) bins may be added together to form a single energy volume.

Finally, subtraction of two volumes reconstructed using different algorithms, or processed with different denoising techniques allows the user to visualise the

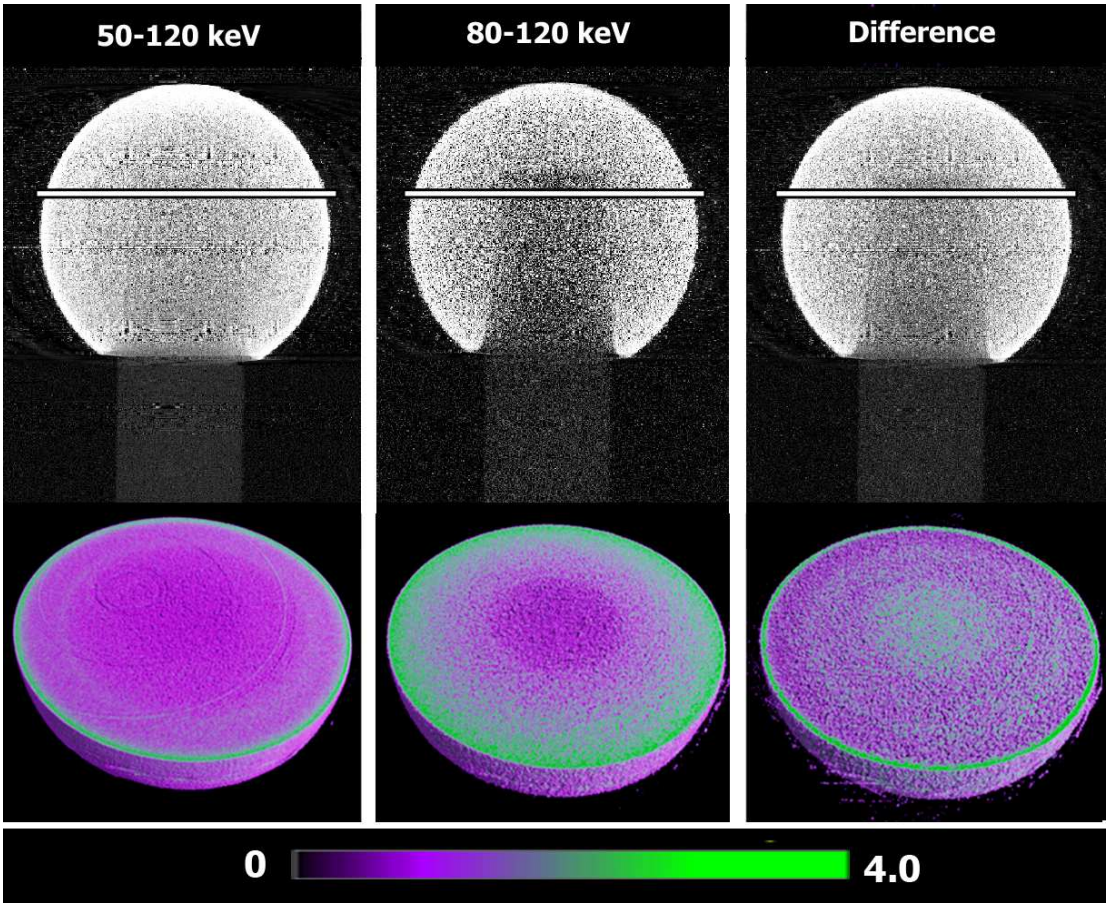


Figure 8.16: Difference between slices from two energy volumes of the CoCr Ball dataset (section 9.5.2), as calculated by MARS Vision's volume data processing algorithms. Top: 2D visualisation. Bottom: DVR with the clipping plane set to show the detail inside the implant. The location of the clipping plane is shown as a line in the 2D slices. The colour bar at the bottom shows the difference in linear attenuation units.

differences between them. This technique has already been used during the development of MARS reconstruction algorithms, but has been restricted to 2D slices [59]. MARS Vision's volume data processing tool allows the users to calculate the difference volume, which can subsequently be shown using either DVR or slice visualisation.

The CoCr Ball dataset can be used as an example. The dataset consists of four energy volumes (50-120, 60-120, 70-120 and 80-120 keV). Differences between two bins can be visualised without any further processing (Figure 8.16 left and middle). Alternatively, a difference volume can be calculated and displayed (Figure 8.16 right). A similar technique, called digital image subtraction, is a currently used in

clinical CT imaging [309, 310]. It involves scanning the patient twice: once before the contrast agent is injected, and once after. The images are then subtracted to show the distribution of the contrast agent.

Overall, this is a very basic data processing tool that has originally been implemented before the introduction of the material volume data format. It is based on the concepts proposed by de Ruiter in his earlier work on spectral CT data visualisation [129]. Currently, this tool is rarely used by the MARS team members, but future work may focus on the development and implementation of a more feature-rich system for processing volume data.

8.4.2 Tools and GUI

The GUI of the volume data processing tool is shown in Figure 8.17. It contains the tools for adding, subtracting, dividing and multiplying volume data, as well as the tools for clamping the entire volume to a certain minimum or maximum value.

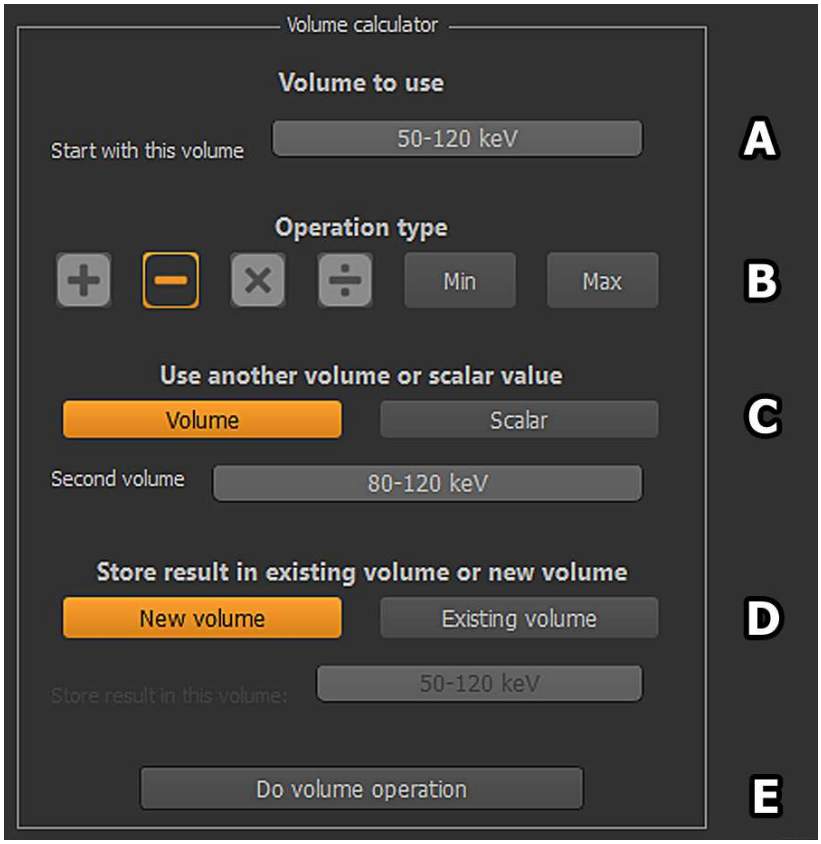


Figure 8.17: MARS Vision's volume data processing GUI.

The GUI breaks down the volume data editing process into four steps, as shown in Figure 8.17:

(A) Selection of the volume to modify (first operand).

(B) Selection of the operation type (operator).

(C) Selection of the second operand. This can either be a different volume (only if the addition, subtraction, multiplication or division operator is chosen), or a scalar value.

(D) Selection of the volume in which to store the results of the operation. Any of the existing volumes may be used; alternatively, a new volume can be created.

Once the "Do operation" button (Figure 8.17E) is pressed, the operation is started. If both operands are volumes, then every voxel of the first volume is modified by the corresponding voxel (that is, the voxel at the same position) of the second volume, depending on the chosen operator. If a scalar value is used, then every voxel of the first volume is changed by that value according to the chosen operator.

8.4.3 Summary

Basic volume data processing tools are mostly used for calculating the difference between two energy volumes of a MARS spectral CT dataset. The difference can be displayed 2D or 3D visualisation techniques.

8.5 Noise and artefact reduction and interactive volume data editing

This section describes the methods of minimising the visual effects of noise and image artefacts, as well as the tools for editing volume data during visualisation. As explained in section 3.3, MARS spectral CT datasets contain significant amounts of noise and various image artefacts that pose a serious problem during visualisation. Removal of these defects can be performed at various stages of the data processing toolchain. For example, the MARS toolchain performs this task both before reconstruction, and after it [59].

Ideally, noise and artefacts should be automatically removed prior to visualisation. However, this is not currently possible, as the causes and solutions to all artefacts are not yet known, and most MARS datasets are affected to some extent. Therefore, visualisation becomes the final stage in the MARS data processing toolchain at which artefacts can be removed.

The methods that can be used for this purpose are also related to another necessary task: the removal or certain regions of the dataset. This may be done for various reasons, for example, to increase the rendering speed, or to prevent occlusion.

Figure 8.18 illustrates a situation where volume data should be edited prior to visualisation. In this case, the calibration tubes that are attached to the scanned object occlude it from almost all camera angles. This is a common problem for the MARS group, as calibration tubes are frequently used to provide examples of known material concentrations to calibrate the MD algorithm [36].

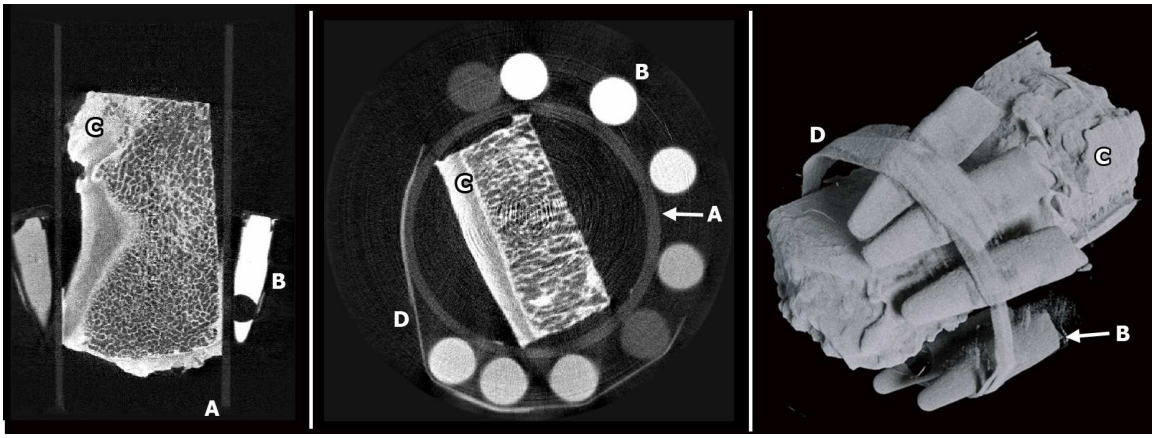


Figure 8.18: Features present in the energy volumes of the Knee Cartilage dataset (section 9.4). **A**: Tube holding the sample. **B**: Calibration tubes. **C**: Sample. **D**: Tape used to attach the calibration tube to the sample tube.

In conclusion, at the visualisation stage, the problems of noise and artefact suppression and removal of unwanted regions of the volume are linked together. The visual effects are similar, although the underlying causes are not. This section shows that the same tools can be used to remove both types.

8.5.1 Cubic B-spline interpolation

The use of cubic B-spline interpolation (described in section 5.3.3.1) is the simplest way of improving the image quality, as no parameters need to be configured, and no other input from the user is required. This algorithms suppresses speckle, or salt-and-pepper noise, as it is less susceptible to sharp changes in the underlying volume data. In addition, small structures, such as streak artefacts in a region of air, are reduced in size.

Cubic B-spline interpolation reduces the performance of the DVR algorithm, as demonstrated in section 5.5.4. Nevertheless, it is a good option for suppressing minor streaks and random noise. The effect of cubic B-spline interpolation is shown in Figure 8.19C and D.

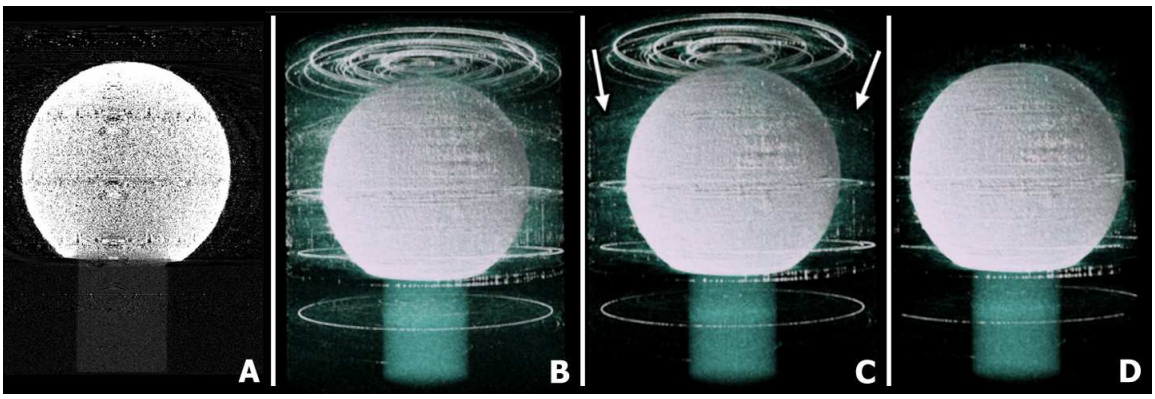


Figure 8.19: Visualisation of the CoCr Ball dataset (section 9.5.2). **A**: 2D slice visualisation. **B**: DVR using trilinear interpolation and no clipping planes. **C**: DVR using cubic B-spline interpolation and no clipping planes. Noise inside the air region is reduced (marked by arrows), but the ring artefacts are not removed. **D**: DVR using both cubic B-spline interpolation and clipping planes to display.

8.5.2 Clipping planes

Clipping planes have been described in section 5.3.6. MARS Vision's DVR algorithm supports clipping along the x, y and z axes. Clipping planes are usually used for revealing occluded regions inside the dataset. However, they can also be used for removing artefacts and other unwanted features.

Clipping planes are axis-aligned and hide all data on one side of the plane. Therefore, they are not a good option if the structures that need to be removed

are geometrically complex. In practice, clipping is most suitable for removing the structures lying on the edges of the dataset, as shown in Figure 8.19D.

8.5.3 2D polygon-based volume cropping

This tool allows the user to remove parts of the volume based on a polygon drawn in any 2D slice view. It can be used in cases when the geometry of occluding structures is too complex to be removed by positioning a clipping plane and when the size is too large to be effectively suppressed by cubic B-spline interpolation.

In these situations, cropping based on an arbitrary polygon mask may be useful. MARS Vision implements this functionality by extending the 2D ROI drawing tool described in section 8.1.

The Knee Cartilage dataset in Figure 8.18 is one example of a MARS spectral CT dataset where this tool can be used to remove occluding structures. Another example is the Mouse0 dataset where all energy volumes and one material volume contain the sample tube, as shown in Figure 8.20.

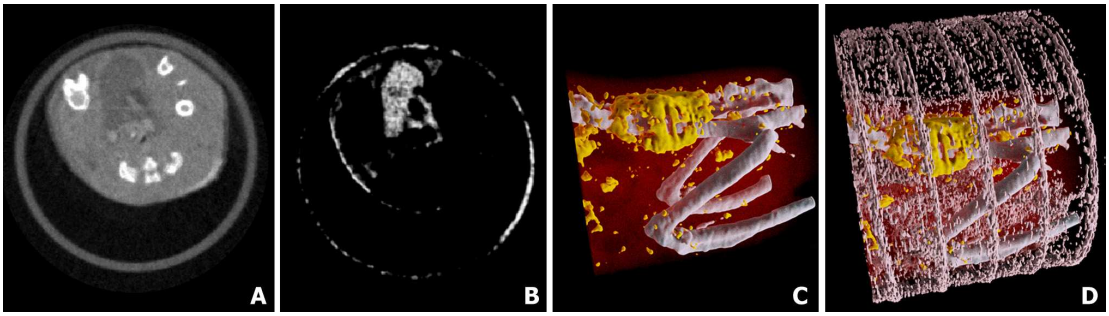


Figure 8.20: Visualisation of the Mouse0 dataset (section 9.6.2). **A**: A slice of an energy volume. **B**: A slice of the lipid material volume. Both slices show the sample tube. **C**: Gold (yellow), calcium (grey) and water (translucent red) volumes visualised using DVR. **D**: Gold, calcium, water and lipid (pink) volumes. The lipid volume occludes the other three materials and makes 3D visualisation ineffective.

The cropping process and its results are shown in Figure 8.21. The use of the polygon cropping tool consists of:

1. Drawing the polygon ROI in any of the 2D slice views (Figure 8.21A and D).

2. Conversion of the drawn polygon into a binary pixel mask. The mask determines which pixels should be retained and which pixels should be removed

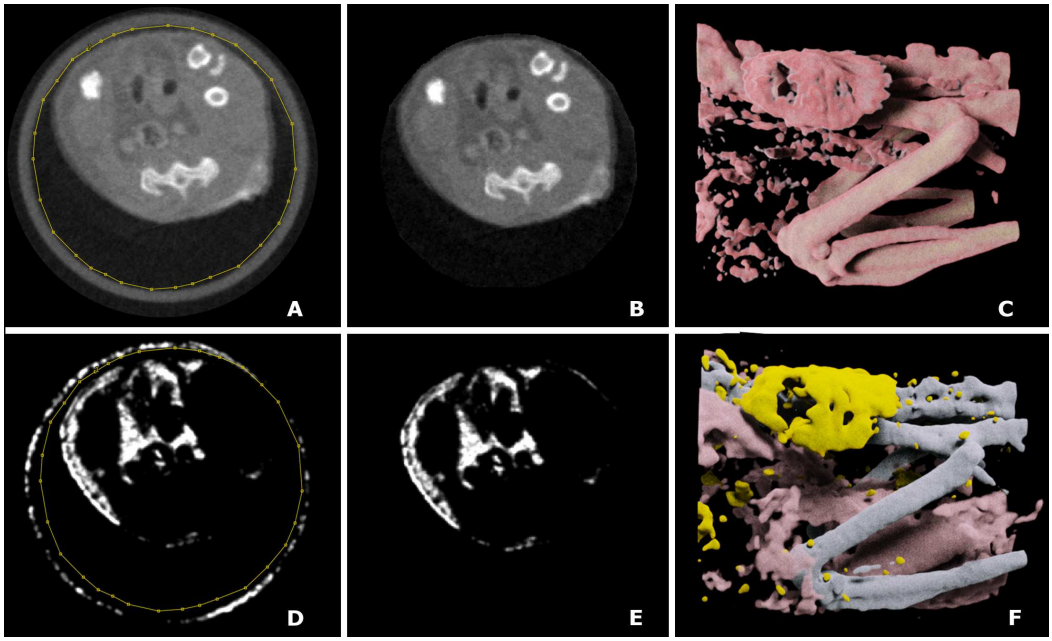


Figure 8.21: Using the 2D polygon-based cropping tool on the Mouse0 dataset (section 9.6.2). **A**: Cropping a slice of an energy volume (outline marked with a yellow polygon). **B**: result. **C**: DVR of the cropped energy volume. **D**: the same crop applied to the lipid volume. **E**: result. **F**: DVR of the cropped lipid volume (pink), with the calcium volume (grey) and the gold volume (yellow) provided for context.

from the volume. The user is able to specify whether the pixels inside or outside the polygon should be removed, and whether a single volume, or all volumes should be cropped.

3. Processing of all slices of the dataset. The slices are processed sequentially in the chosen direction (axial, sagittal, or coronal). The mask is applied to every slice. if the pixel is inside the mask, it is removed (value is set to the minimum of the volume's data range).

Once all slices have been processed, the volume is updated and rendering is restarted. The modified volume or volumes can either be saved to disk, or discarded when MARS Vision is closed.

8.5.4 3D volume editing

The final method of editing volume data relies on the depth buffer, described in section 5.4. If this tool is used, clicking on any visible surface will remove a cube of

voxels around the selected 3D volume coordinate. Similar to the polygon cropping tool, all voxels in the removed block are set to the minimum data value. This process is shown in Figure 8.22.

The utility of this tool can be demonstrated using the following example. As shown in Figure 8.23A, the iodine volume of the Mouse12 dataset contains several regions where iodine was improperly injected. These regions are of no interest to researchers working with this dataset and may obscure the ROI inside the chest of the mouse. The geometry of these structures is too complex to be removed by adjusting the clipping planes or by using the 2D polygon cropping tool.

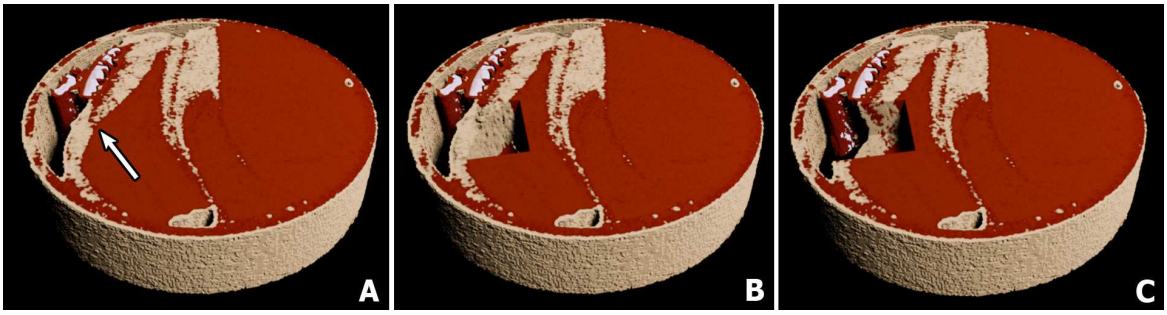


Figure 8.22: The use of the 3D volume data editing tool. **A**: clicking the visible surface at the point marked by an arrow. A block of voxels centred on the chosen coordinate is then cut out from one volume (**B**), or from all volumes (**C**).

Instead, the 3D volume editing tool can be used to directly remove these structures by clicking on the visible surfaces (Figure 8.23B). The final result is shown in Figure 8.23C. All unwanted structures have been removed.

The 3D volume editing tool can be configured to only remove voxels from a single volume, as demonstrated in Figure 8.23, or from all volumes at once, as shown in Figures 8.22C and 8.24. This is useful for editing energy volumes, which may contain image artefacts (such as rings or streaks) in similar locations.

This tool is a very basic solution to the problem of interactive spectral CT volume data editing. In the future, it will likely be extended by incorporating region growing segmentation (section 8.3), where the user sets the initial seed point by clicking on a visible structure, and the segmentation algorithm extracts and removes a 3D region grown from this point.

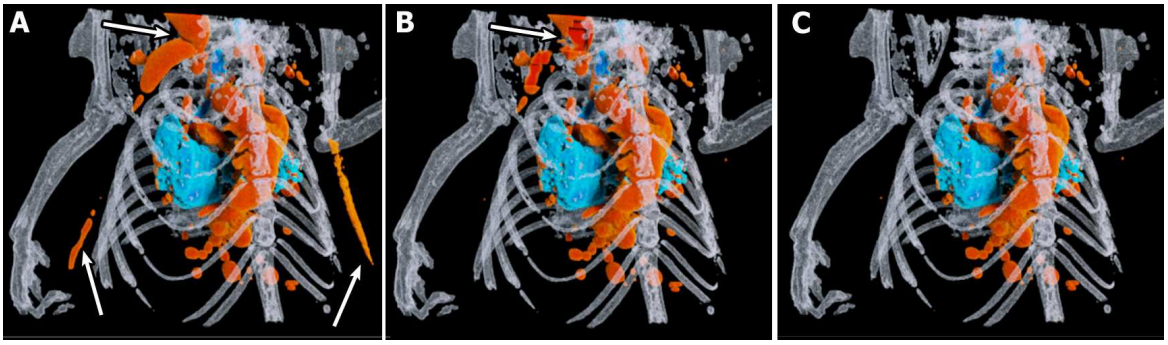


Figure 8.23: Editing of the Mouse12 dataset (section 9.6.1) during 3D visualisation. **A**: original state of the dataset. The unwanted regions of iodine (shown in orange) are marked by arrows. **B**: Intermediate editing stage. Two of the regions have been removed, while the third is still being edited. **C**: Final result.

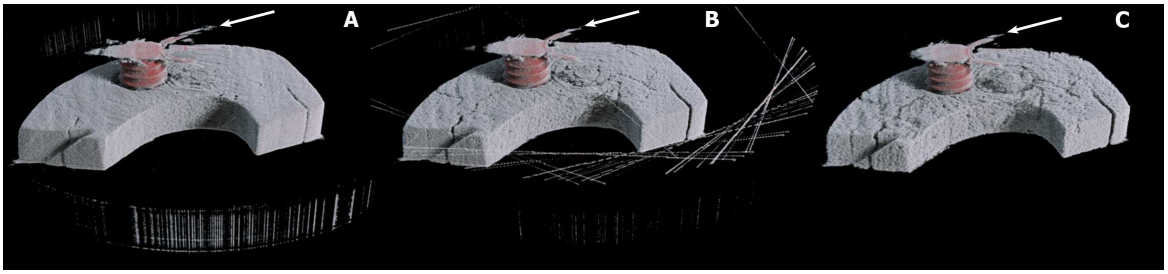


Figure 8.24: Artefacts in three energy volumes of the TiScrew dataset (section 9.5). **A**: 40-120 keV. **B**: 50-120 keV. **C**: 80-120 keV. All volumes contain the same artefact (marked with an arrow).

8.5.5 Summary

The tools for editing volume data, suppressing noise and removing image artefacts are an important addition to MARS Vision. Current MARS spectral CT datasets contain a large amount of artefacts and other unwanted structures that obscure ROIs during 3D visualisation. A combination of cubic B-spline interpolation to suppress speckle noise, clipping planes to remove obscuring regions, and 2D and 3D volume editing tools increases the range of options provided to the user.

8.6 Summary

Previous chapters have shown that the visualisation of spectral CT datasets benefits from high-quality DVR, 2D slice data fusion, custom GUI elements for designing transfer functions and tools for reducing occlusion. However, visualisation

alone is usually insufficient to effectively analyse a spectral CT dataset.

This chapter has described the implementation of some of the tools that are required at the current stage of development of spectral CT technology. These tools facilitate working with MARS spectral CT datasets.

- Measurements of size and composition of ROIs are essential for analysing spectral CT datasets. MARS Vision extends traditional tools by allowing simultaneous measurement of multiple volumes. In addition, a novel tool, called a concentration-volume histogram, can be used to assess the spread of a material inside a chosen ROI.

- Generic presets applicable to all or most MARS spectral CT datasets cannot currently be created due to the lack of standardisation. Instead, MARS Vision supports presets that are specific to a particular dataset. Each dataset-specific preset saves the camera and lighting parameters, all transfer functions, and all annotations and measurements performed by the user.

- Threshold-based segmentation can be used to extract ROIs by growing a volume from a selected seed point, or by partitioning the entire volume. This process creates a new volumetric dataset, which can subsequently be visualised using the magic lens and overlay tools. The original unsegmented volume can be used as the context, while the extracted volume can be shown as an overlay, or inside a magic lens.

- Volume data processing tools allow users to compare two volumes by subtracting them and visualising the result using 2D or 3D techniques. Other actions, such as adding two volumes, or removing all voxels above or below a certain threshold, are also supported. These tools allow advanced users to edit volume data during visualisation, as opposed to performing processing using a separate application prior to visualisation.

- Noise, image artefacts, and other unwanted structures are commonly present in MARS spectral CT datasets. These features can be permanently removed through the use interactive 2D or 3D volume data editing tools or temporarily hidden using clipping planes. These tools are essential, as 3D visualisation is severely disrupted by many artefacts commonly found in currently-available datasets.

Chapter IX

Visualisation of spectral CT datasets with MARS Vision

This chapter describes the workflow supported by the combination of tools developed over the course of this research. The utility of the visualisation algorithms, transfer function editors, and tools described in Chapters 5-8 is tested using a number of MARS spectral CT datasets.

Previous chapters have treated each problem currently facing spectral CT visualisation (for example, occlusion, image artefacts, or the tools for performing data fusion) separately. This chapter discusses the overall workflow and the use of a combination of tools for achieving effective visualisation of a variety of MARS spectral CT datasets.

Six imaging applications investigated by the MARS project are discussed, and the most appropriate visualisation tools to be used for each application are explained. The datasets described in this chapter have also been used in several publications [2, 4, 9, 11, 13, 155]. My contributions to these publications are mentioned when appropriate.

The six dataset types are: phantoms (section 9.1), atherosclerotic plaques (section 9.2), lamb meat tissue samples (section 9.3), excised cartilage samples (section 9.4), metal implants (section 9.5), and small animals (section 9.6).

Finally, the applicability of the tools created over the course of this research to future human-scale spectral CT datasets is discussed in section 9.7. The MARS system cannot currently be used to scan patients, which means that a human-scale spectral CT dataset must be simulated. To do this, a single-energy CT dataset was segmented (to approximate the identification of a material with MD) and visualised using MARS Vision.

9.0.1 A note on material volume data formats

The material volumes created by MARS MD algorithms store data in two formats: relative and absolute concentration. The MD algorithms used at the beginning of

this research did not allow for precise material quantification. Therefore, material concentration was expressed as a fraction relative to the concentration used for calibration. Typically a tube containing a solution with the material of interest as a primary component was attached to the sample.

For example, a relative concentration of 1 means that a material is the same concentration as the reference material, while a relative concentration of 1.5 means that the identified material is 1.5 times more concentrated than the reference material.

By the end of this research, MD advanced to the point where absolute concentration became routinely used. Material volumes stored in this format use mass concentration units (usually g/ml or mg/ml).

9.1 Phantom imaging

In medical imaging, a phantom (also called an imaging phantom) is a specially-prepared object used for calibrating acquisition hardware, testing reconstruction software, or evaluating the performance of an imaging system in general.

Phantoms are excellent for testing because their dimensions, geometry, and composition can be precisely controlled. This means that once a phantom sample is scanned, the data (for example, raw projection frames or reconstructed volumes) can be compared to the expected result, anomalies can be detected, and necessary adjustments can be made.

Phantoms can be used for research or teaching purposes, and can be constructed for all common medical imaging modalities: for example, CT [311], PET/CT [312], SPECT [313], [314], MRI [315], ultrasound [316], and EEG [317]. They are often used to test the experimental scanner hardware that may be hazardous to human or animal subjects.

In addition, phantoms are an essential part of scanner maintenance. For radiation-based modalities, such as CT, phantoms are used to test dosimetry (to meet legal requirements), and to calibrate the response of the system so that results are consistent across scans. Multiple scans of the same phantom should always produce the same results; if not, then the technical staff will adjust the scanner's settings to ensure that this is the case.

All MARS scanners are first tested using phantoms, before being used to scan tissue samples, or live animal specimen. A number of other research teams de-

veloping prototype spectral CT systems have used phantoms to test the scanner hardware, and reconstruction and MD techniques. For examples, refer to the studies by Cormode et al. [5], Boll et al. [12], Feuerlein et al. [7] and Shikhaliev [221].

9.1.1 Spectral calibration phantom

The MARS team uses spectral calibration phantoms (also known as multicontrast phantoms) to determine how well materials of interest can be separated, and to calibrate the material decomposition algorithms.

All spectral calibration phantoms used by the MARS team are similar in shape, but contain different concentrations of materials. Usually, such phantoms are small PMMA (polymethyl methacrylate, also known as Perspex) cylinders around 5 cm in diameter, with several tubes containing solutions of different materials. An example is shown in Figure 9.1.

For the purposes of testing the visualisation tools, examining one spectral calibration phantom is sufficient. Basic information about the Spectral Phantom dataset is shown in Table 9.1.

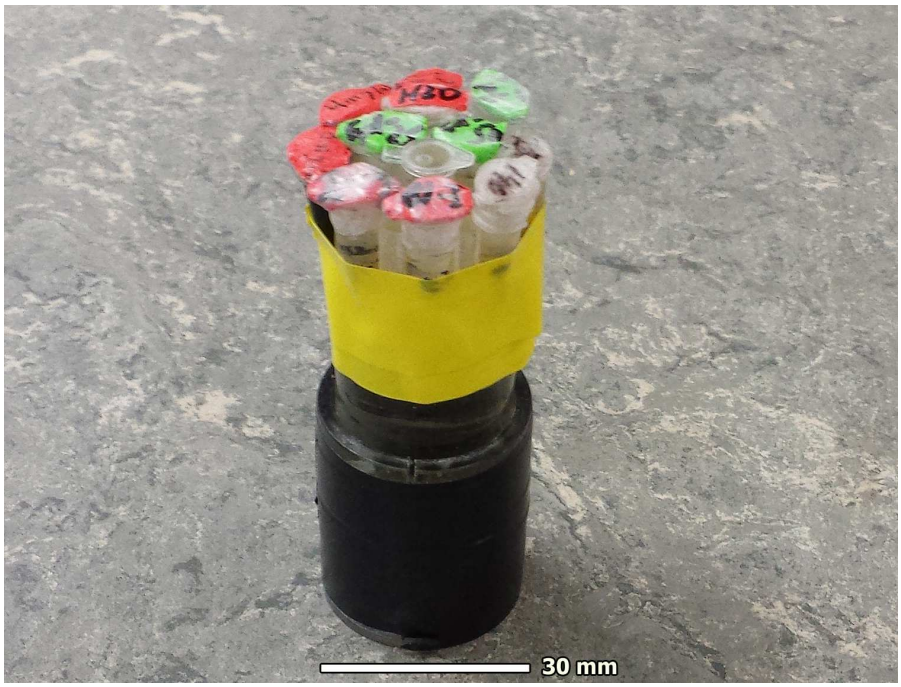


Figure 9.1: Photograph of a spectral calibration phantom used for calibrating MARS MD algorithms. This is not the same phantom visualised in Figure 9.2, although the two are nearly identical in shape in size.

Table 9.1: Basic information about the Spectral Calibration Phantom dataset.

Parameter	Description
Name	Spectral Phantom
Year acquired	2014
Objects scanned	Imaging phantom containing aqueous solutions of different concentrations of calcium, iodine, gadolinium and gold.
Study objective	Calibration of MARS MD algorithms and evaluation of their accuracy and sensitivity.
Energy volumes	8
Material volumes	5 (water, gadolinium, iodine, gold, calcium)
Data format(s)	Linear attenuation coefficients (energy volumes), absolute concentration (material volumes)
Dataset dimensions	$246 \times 246 \times 16$ voxels
Bits per voxel	16

The phantom shown in Figure 9.2 contains a set of solutions of materials over a range of concentrations. Materials include gadolinium (Magnevist: 2.1, 4.2 and 8.4 mg(Gd)/ml), calcium (CaCl: 140 and 280 mg(Ca)/ml), gold (gold nanoparticles: 2, 4 and 8 mg(Au)/ml), iodine (Omnipaque: 9, 18 and 36mg(I)/ml), and water. Solutions are held in a PMMA frame.

MD has been used to identify and quantify five materials, which correspond to the four materials of interest (gold, gadolinium, calcium, iodine), and water. Visualisation of these material channels is trivial. The simple transfer function editor (Figure 9.2H) can be used to generate a colour gradient for each material.

In this case, the only requirement is that the colours for the different material volumes are distinct. Gadolinium ranges from light to dark green, calcium ranges from orange to red, gold ranges from light to dark yellow, iodine ranges from cyan to blue, and water is assigned a light grey colour. Note that the attenuation of PMMA is similar to water, and that PMMA was deposited in the water volume as it was not a target material for separation in material decomposition.

The only overlap between the materials can be found between the water channel, and every other material. This is expected, as this phantom contains aqueous solutions of gold, calcium, gold, and iodine, which means that each voxel containing some amount of any of these materials will also contain a certain amount of water.

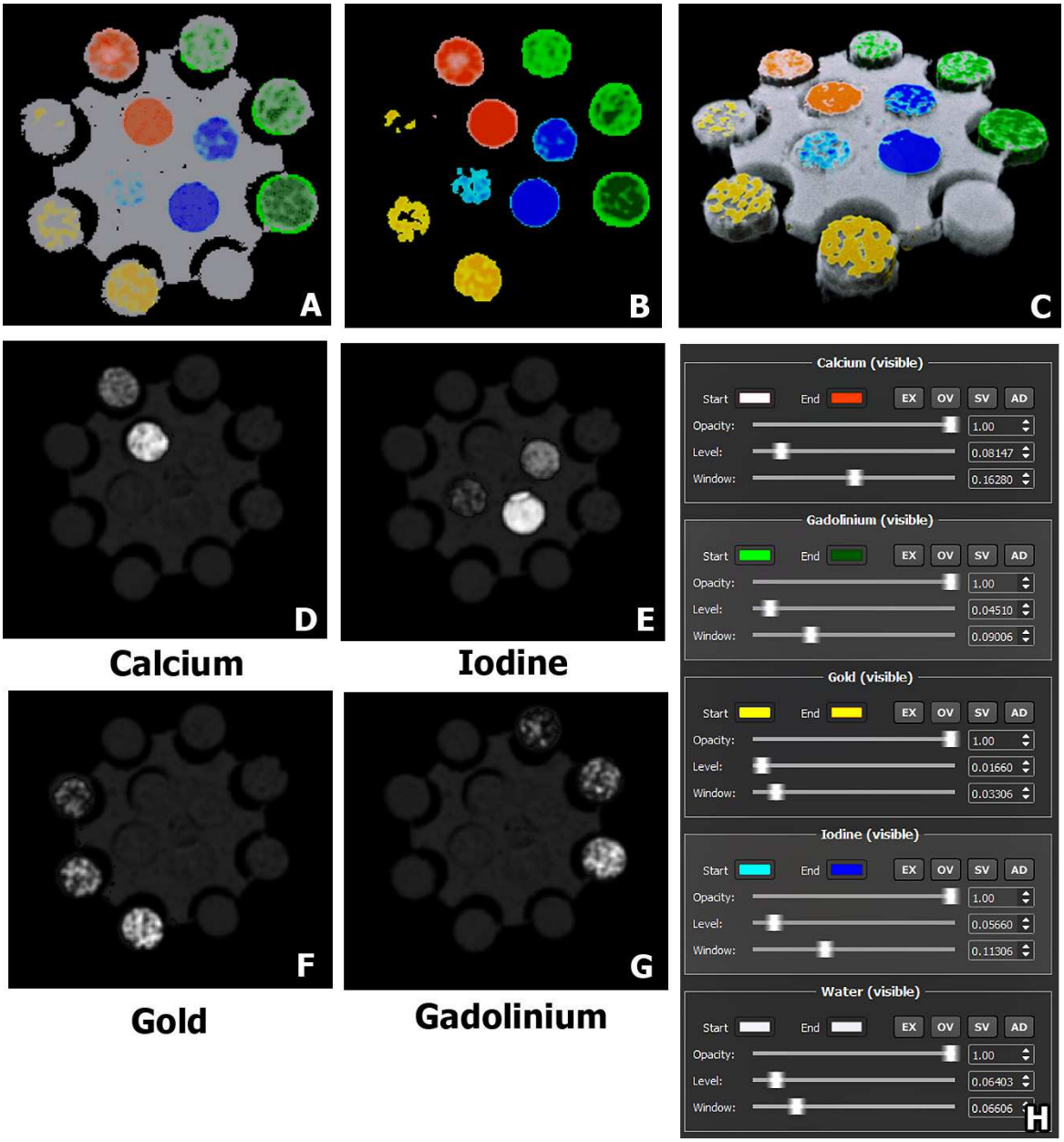


Figure 9.2: Visualisation of the Spectral Phantom dataset. **A**: Spectral mode data fusion of all material channels. **B**: Gold, calcium, gadolinium, and iodine material channels. **C**: DVR of all material channels. **D**-**G**: individual material channels. The outline of the water channel (dark grey) has been added manually in post-processing to provide context. **H**: the transfer functions used in images **A**-**C**.

Overall, this dataset, along with all other calibration phantoms, does not present an interesting challenge, as occlusion is not a problem, and the geometry of the object is very simple. The only interesting aspect of multricontrast

phantom visualisation is the use of 2D spectral mode data fusion. In this case (Figure 9.2A), the grey colour assigned to the water channel interferes with the visualisation of the colour gradient for all other materials. This is easily remedied by hiding the water channel (Figure 9.2B).

As mentioned above, this phantom can be used for calibrating MD algorithms. Therefore, visualisation, and especially 3D rendering, is unlikely to be performed by users. However, in this thesis, this dataset can be used for effectively demonstrating simultaneous 2D and 3D visualisation of multiple volumes, as well as for demonstrating the use of the simple transfer function editor for material volumes.

9.1.2 Summary

Phantoms are essential for testing the capabilities of medical imaging equipment. The MARS project uses a number of spectral calibration phantoms to obtain attenuation values corresponding to known concentrations of materials. Phantoms are excellent for showing the material discrimination capabilities of the MARS system, as the detected range of concentrations for each material can be clearly visualised.

The simple transfer function editor is ideally suited for working with spectral calibration phantoms. Each material is associated with a particular colour gradient, while the PMMA frame and the water channel can be assigned a dark colour, or hidden completely.

In the future, spectral calibration phantoms are unlikely to be visualised directly. However, currently, they provide an opportunity to test the capabilities of the visualisation tools developed over the course of my research. For example, the colour distortion as a result of blending with the background material, and the use of the transfer function editor to assign colour gradients to material channels can be demonstrated.

9.2 Atherosclerotic plaque imaging

Phantom studies are a fast, cheap and effective method of assessing the performance of the MARS system, but the majority of pre-clinical research carried out by the MARS project involves real biological tissues. The first application that involves biological tissue is the imaging of atherosclerotic plaque.

Atherothrombotic cardiovascular disease (ACD) is one of the leading causes of death in New Zealand and many other countries [318, 319]. Atherosclerosis is the cause of ACD, along with a number of other conditions, such as stroke and peripheral vascular disease. Atherosclerosis is an inflammatory process that leads to the thickening of the artery walls and the formation of atherosclerotic plaques. This process is shown in Figure 9.3.

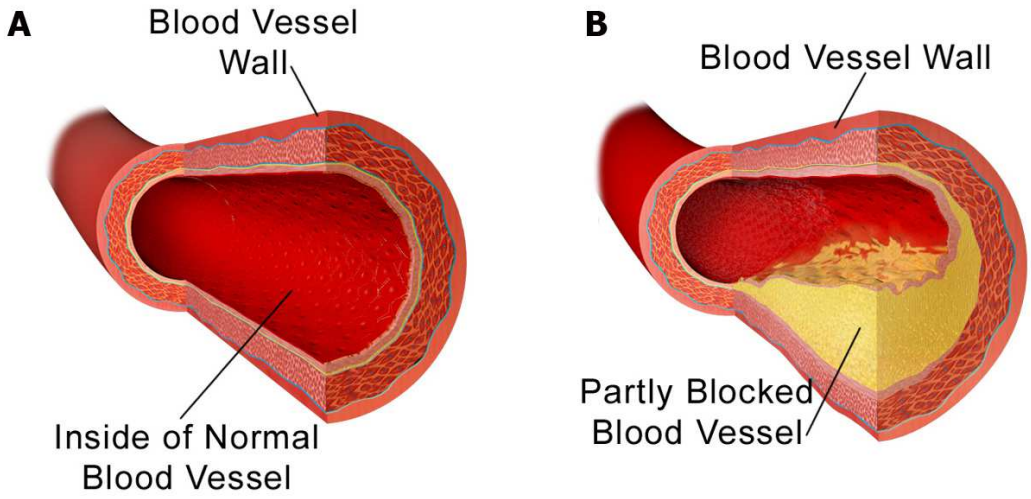


Figure 9.3: The effects of atherosclerosis. **A**: Healthy artery. **B**: Thickened artery wall due to the inflammation caused by atherosclerosis. ©Blausen Medical Communications, Inc., used under the terms of the CC-BY-3.0 license.

At first, atherosclerotic plaques are aggregations of lipid-rich white blood cells, but, as the diseases progresses, some regions may start to calcify [320]. Therefore, the structure and composition of a plaque may be assessed by measuring both the lipid content, and the size and location of calcifications. This information could potentially be used to classify the *vulnerability* of a plaque (how prone it is to rupturing).

Research into plaque vulnerability is essential, as rupture of unstable plaques is the cause of around 70% of heart attacks [321]. Spectral CT is one of the medical imaging technologies that can be used for this purpose [2].

9.2.1 Plaque 72 and Plaque77

The Plaque72 and Plaque77 datasets are results of some of the early studies into atherosclerotic plaque imaging conducted by the MARS team. Basic information about these datasets is summarised in Table 9.2.

Table 9.2: Basic information about the Plaque72 and Plaque77 datasets.

Parameter	Description
Name	Plaque72 and Plaque77
Year acquired	2013
Objects scanned	Excised sections of the human carotid artery
Study objective	Assessment of the composition of atherosclerotic plaque using spectral CT
Energy volumes	4
Material volumes	3 (water, lipid, calcium)
Data format(s)	Linear attenuation coefficients (energy volumes), relative concentration (material volumes)
Dataset dimensions	$583 \times 583 \times 99$ voxels (Plaque72), $269 \times 269 \times 224$ voxels (Plaque77)
Bits per voxel	16

The Plaque72 dataset has been described in the paper by Panta et al. [11]. My contribution to this work involved generating the images showing the distribution of water (corresponding to soft tissues), lipid (corresponding to fat tissues), and calcium inside this plaque. These colours have been chosen based on the feedback from R. Panta, the primary investigator of this study. The requirements were:

1. The colour schemes used for different materials should be distinct.

2. The chosen colour gradients should show the concentration gradients (if any are present).

3. The colours should be reasonably close to the colours of human body tissues to help the users interpret the images.

In addition to working with Panta et al., I have provided two images that have been used in the review paper by Anderson and Butler [2]. These images demonstrate the ability of the MARS system to discriminate between the materials comprising an atherosclerotic plaque.

Structurally, both datasets are very similar, as they show the *bifurcation*, or the point where the artery splits into two separate blood vessels. In the case of the carotid artery, as shown here, one of these vessels supplies the neck, while the

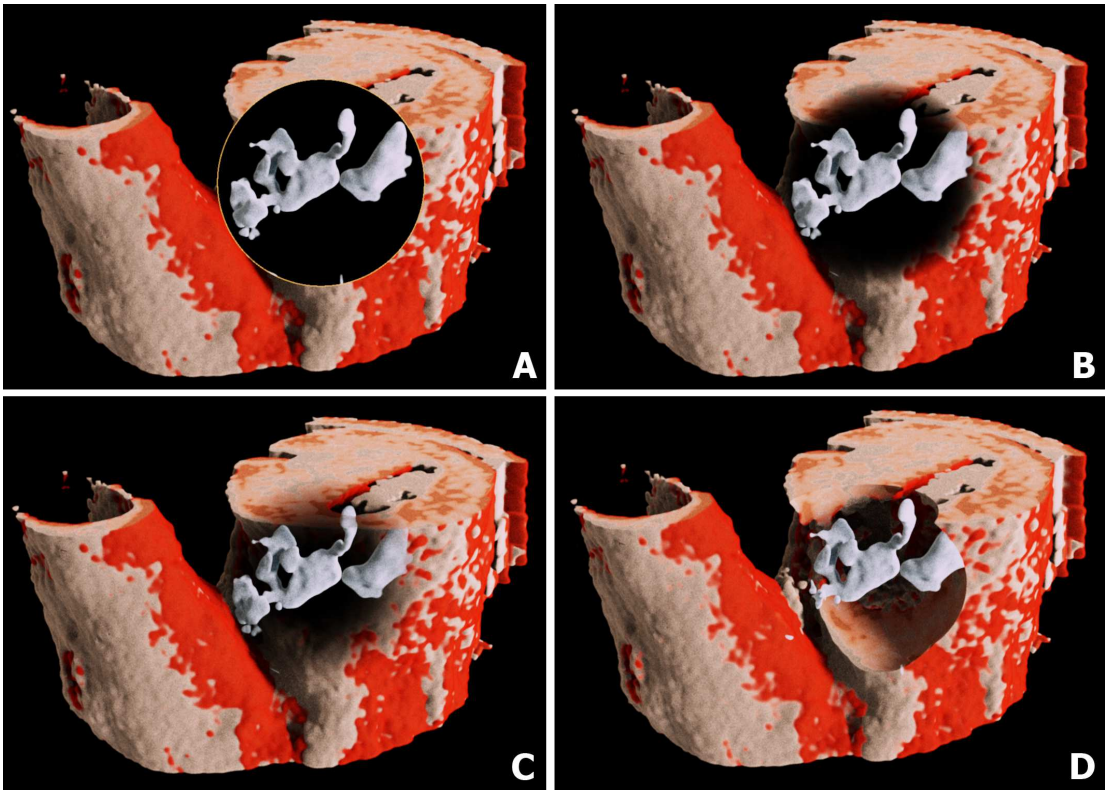


Figure 9.4: Use of magic lenses to visualise the location of calcifications inside the Plaque72 dataset. **A**: 2D magic lens. **B**: 2D magic lens with low context preservation. **C**: 2D magic lens with high context preservation. **D**: 3D magic lens. Calcium is white, lipid is beige, and water is red.

second supplies the brain with blood. From the point of view of visualisation, both datasets consist of the lipid and water volumes that overlap in multiple places, with small masses of calcium (calcifications) interspersed throughout.

From the observations made over the course of this research, this appears to be a common situation in atherosclerotic plaque imaging. It has been one of the key reasons for developing the tools to clearly visualise small ROIs occluded by a large mass of soft tissue. Figure 9.4 shows the use of the magic lens tool for displaying the location of the calcifications along with the context. This is necessary since, by themselves, the calcifications do not contain enough contextual information, which is essential for visualising their location in relation to the soft tissue.

2D and 3D data fusion can be used to clearly visualise the location of the lipid and water channels, as shown in Figure 9.5. Images similar to the one shown in Figure 9.5B are included in the papers by Panta et al. and by Anderson and Butler.

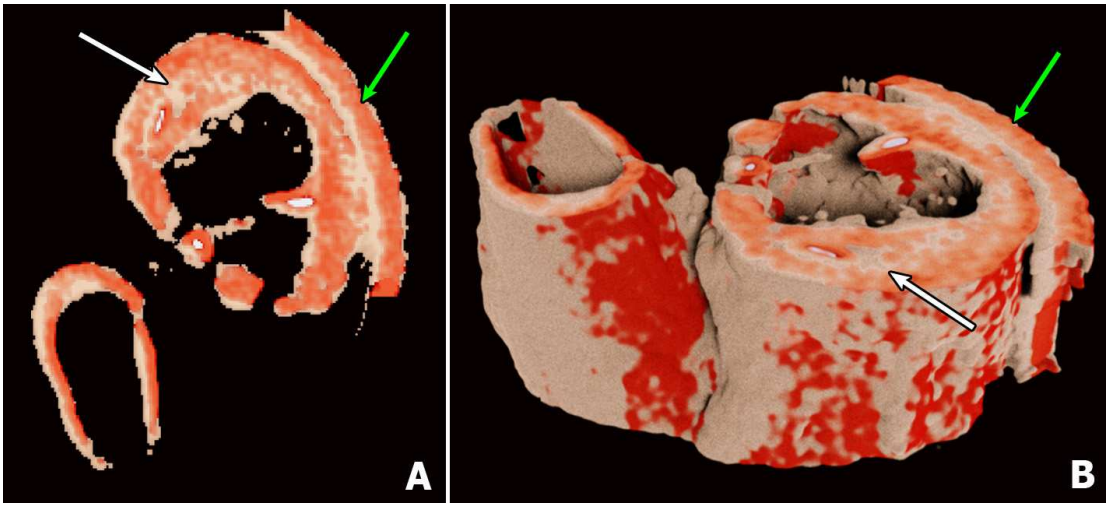


Figure 9.5: **A**: 2D spectral mode data fusion of the three material channels of the Plaque72 dataset. **B**: DVR of the three materials, with the clipping plane set to show the same slice. The lipid-rich core of the plaque is marked by white arrows. The segment of the Perspex sample tube that has not been cropped out is visible on the right side of both images (marked by green arrows). The entire sample tube can be seen in Figure 9.6.

Displaying the entire plaque using DVR is useful for conveying the information about its overall structure to external researchers, who are not familiar with this dataset. Traditional slice visualisation can still be used to clearly show specific ROIs.

During DVR, the brightness of the directional light can be lowered to reduce the intensity of the shadows (Figure 9.5B) and minimise the colour distortion caused by lighting. In both cases, blending between the material channels occurs naturally as a result of the colour blending algorithms implemented in MARS Vision.

Visualising the three materials together with a single energy volume allows us to use the technique of material overlay. Material overlay, using the 2D colour replacement mode, is shown in Figure 9.6. This figure which also shows the sample tube, which has not been cropped out in this image; the tube is mostly removed (using the 2D polygon crop tool) in Figures 9.4 and 9.5.

Finally, Figure 9.7 shows how a colour gradient created with the simple transfer function editor can be used to visualise the concentration gradient in a single material channel. In this case, unrealistic colours have been deliberately used to emphasise the differences in concentration. I have rendered and provided a similar image to be used in the review paper by Anderson and Butler [2]. The image

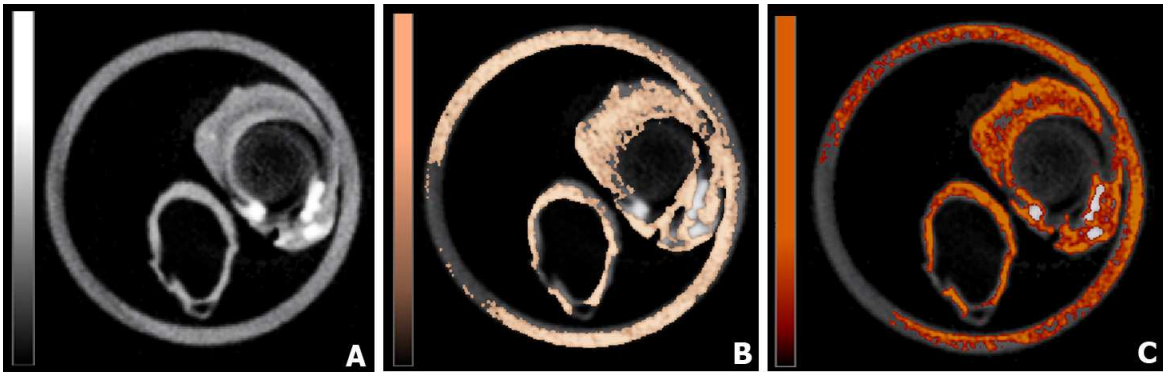


Figure 9.6: **A**: A slice of an energy volume from the Plaque72 dataset. **B**: Overlay by the lipid volume. This is achieved by concurrently visualising the lipid volume and the energy volume, and by using the colour replacement mode (section 5.6.4.1) to preserve the colour gradient for the lipid volume. **C**: Overlay by the water volume. Note the presence of the sample tube, which has not been cropped out. Sample tubes do not interfere with slice visualisation, and therefore do not need to be removed.

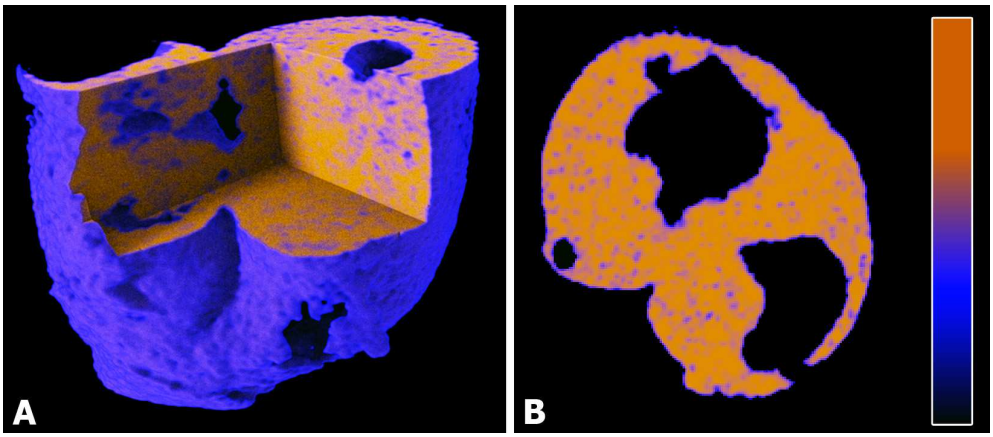


Figure 9.7: Visualisation of the changes in the concentration of the lipid channel of the Plaque77 dataset. **A**: DVR with a clipping box. **B**: 2D spectral mode slice visualisation. The colour gradient generated by the simple transfer function editor is shown on the right.

shows the concentration gradient of lipid inside the Plaque72 dataset.

The tools described above can be used to identify regions of interest inside these plaques, such as the lipid-rich core (highlighted in Figure 9.5) which is currently thought to be a sign of a plaque that is prone to rupturing [322]. Once the lipid core, the location of the calcifications, or other features of interest are identified, they can be analysed in greater detail. Preliminary research by the MARS team has demonstrated the potential use of spectral CT for analysing the structure and composition of atherosclerotic plaques, but further research is required.

9.2.2 Plaque 108

This is a recently-acquired atherosclerotic plaque dataset, scanned in early 2015. Basic information about this dataset is summarised in Table 9.3. It is different from Plaque72 and Plaque77, as the amount of calcium is much greater. Calcium is not distributed in small granules inside the soft tissue, but occupies a large volume of space. This is shown in Figure 9.8. For consistency, this visualisation uses a colour scheme that is similar to the one applied to the earlier Plaque 72 dataset.

Table 9.3: Basic information about the Plaque108 dataset.

Parameter	Description
Name	Plaque108
Year acquired	2015
Objects scanned	Excised section of the human carotid artery
Study objective	Assessment of the composition of atherosclerotic plaque using spectral CT
Energy volumes	4
Material volumes	3 (water, lipid, calcium)
Data format(s)	Linear attenuation coefficients (energy volumes), relative concentration (material volumes)
Dataset dimensions	$384 \times 384 \times 288$ voxels
Bits per voxel	16

The workflow is similar to the other two plaque datasets. First, the sample tube needs to be cropped out to provide an unobstructed view of the plaque. Next, the colour gradients are assigned to differentiate between the material channels. The simple transfer function editor is sufficient, as shown in Figure 9.8. Finally, the clipping planes or the clipping box (Figure 9.8B) can be used to study the structures inside the plaque.

The magic lens tool is not useful in this case, as the volume of space occupied by calcium is large relative to the other two materials. However, the threshold intensity projection can be used to display the outline of calcium together with the context. The context can be provided by either the lipid, or the water volume, or by both volumes simultaneously.

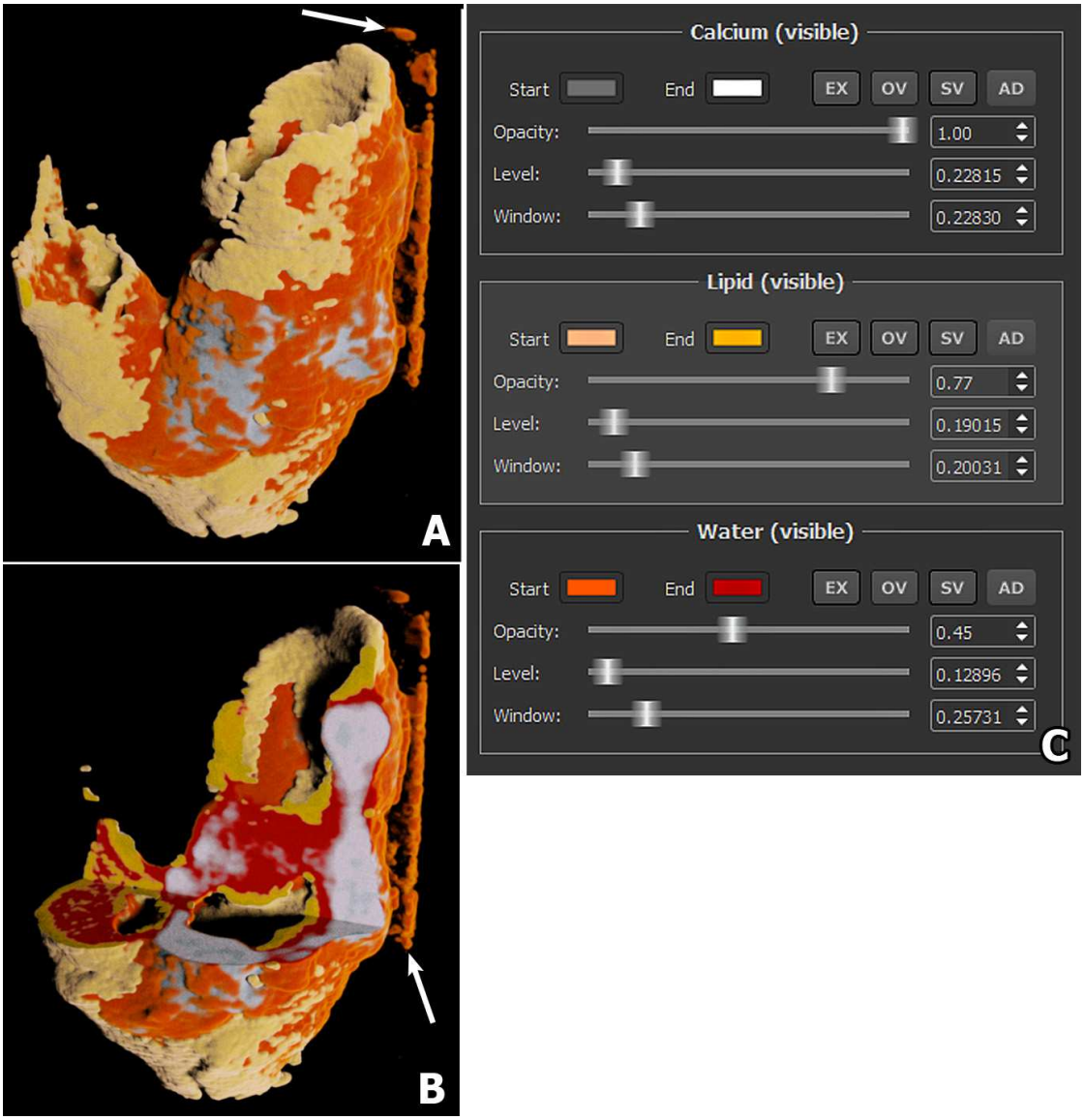


Figure 9.8: DVR of the Plaque108 dataset with MARS Vision. **A**: Entire dataset. **B**: the clipping box used to reveal the distribution of calcium inside the plaque. Calcium is white, water is orange, and lipid is yellow. The sample tube has not been cropped out completely; its remnants are marked by arrows. **C**: the simple transfer function editor settings used to generate the colour gradients for the three material channels.

TIP is intended to be used interactively, and its utility is difficult to show in a static image. Figure 9.9 renders the Plaque108 dataset from two different angles to show how the user may perceive the depth information by rotating the camera.

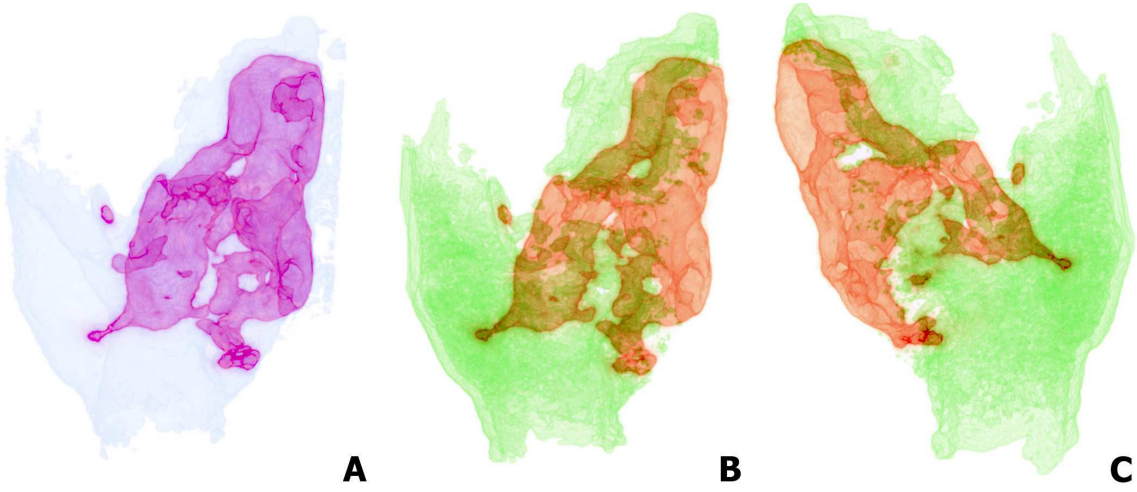


Figure 9.9: TIP rendering of the the Plaque108 dataset. **A**: calcium (red/pink) shown together with the water channel (purple). The emphasis mode is used to reduce the intensity of the colour assigned to the water channel. **B**, **C**: calcium shown together with the lipid channel (green) from two different camera positions.

9.2.3 Summary

In atherosclerotic plaque datasets, the important ROIs are generally the lipid cores and the calcifications. The lipid cores can be visualised by assigning a colour gradient to the lipid and the water channels, and by searching for the locations where the two materials do not overlap. In MARS Vision, this can be done by using the simple transfer function editor, and by using clipping planes to reveal the structure inside the dataset. The location of calcifications can be clearly visualised by using the same technique, or with the help of the overlay, magic lens or TIP tools, depending on the case.

In conclusion, 3D visualisation of atherosclerotic plaque dataset is not a difficult problem. It requires a combination of the polygon cropping tool (to remove the sample tube), standard 2D or 3D visualisation, magic lenses and, sometimes, TIP. The simple transfer function editor is sufficient, as energy volumes are usually not studied directly.

9.3 Lamb meat imaging

Studying the application of spectral CT to imaging of lamb meat is important for several reasons. First, it is a good human tissue analogue, and can be scanned

to evaluate the ability of the MARS system to differentiate between muscle and fat tissue. On a molecular level, human tissues are largely composed of water, with other materials, such as lipids and calcium also being present [323]. MARS MD (section 3.2.2) identifies water-like and lipid-like channels, which largely correspond to muscle and fat tissues. This capacity for precise material identification is the reason why spectral CT is expected to improve upon currently-employed techniques for soft-tissue imaging [2].

Second, it is a potential future industrial application of spectral CT. Automated CT scanning and analysis of meat can be used to determine its fat distribution and assess its quality, as well as prepare the carcass for automatic cutting and processing [324, 325].

9.3.1 Meat1127

This section describes the lamb meat dataset acquired in 2013 by Aamir et al. [4]. My contribution to this paper involved preparing the energy and material volumes for visualisation, creating transfer functions and generating all 3D renderings used in the paper.

For this study, the MARS system was used to scan a small piece of lamb meat in order to quantify its fat content, and determine whether the fat distribution patterns known as "marbling" can be identified. MD of the Meat1127 dataset has produced three volumes: water-like tissues (corresponding to muscle tissue), fat-like tissues, and calcium-like tissues (bones). The colour schemes for the three material channels were chosen to approximately match the colours of bone, muscle, and fat tissue. Basic information about this dataset is summarised in Table 9.4.

The positions of the features of interest (marbling patterns) can be displayed using both 2D, and 3D visualisation, as shown in Figure 9.10. While small, these patterns are clearly present in the lipid material channel. Note that, in this case, the tube holding the sample has not been removed, as it does not interfere with 3D visualisation. In addition, TIP can also be used to show marbling patterns. This has been demonstrated earlier, in Figure 7.10.

9.3.2 Summary

2D, DVR and TIP visualisation techniques implemented in MARS Vision can be used to show the distribution of marbling patterns in the meat datasets pro-

cessed by the MARS molecular imaging system. The process of visualising of these datasets is in many ways similar to the method for visualising atherosclerotic plaques. In both cases, the datasets consist predominantly of soft and fat tissues, with small masses of calcium inside. TIP can be used to show the outlines of structures of interest (in the case of the Meat1127 dataset, the marbling patterns).

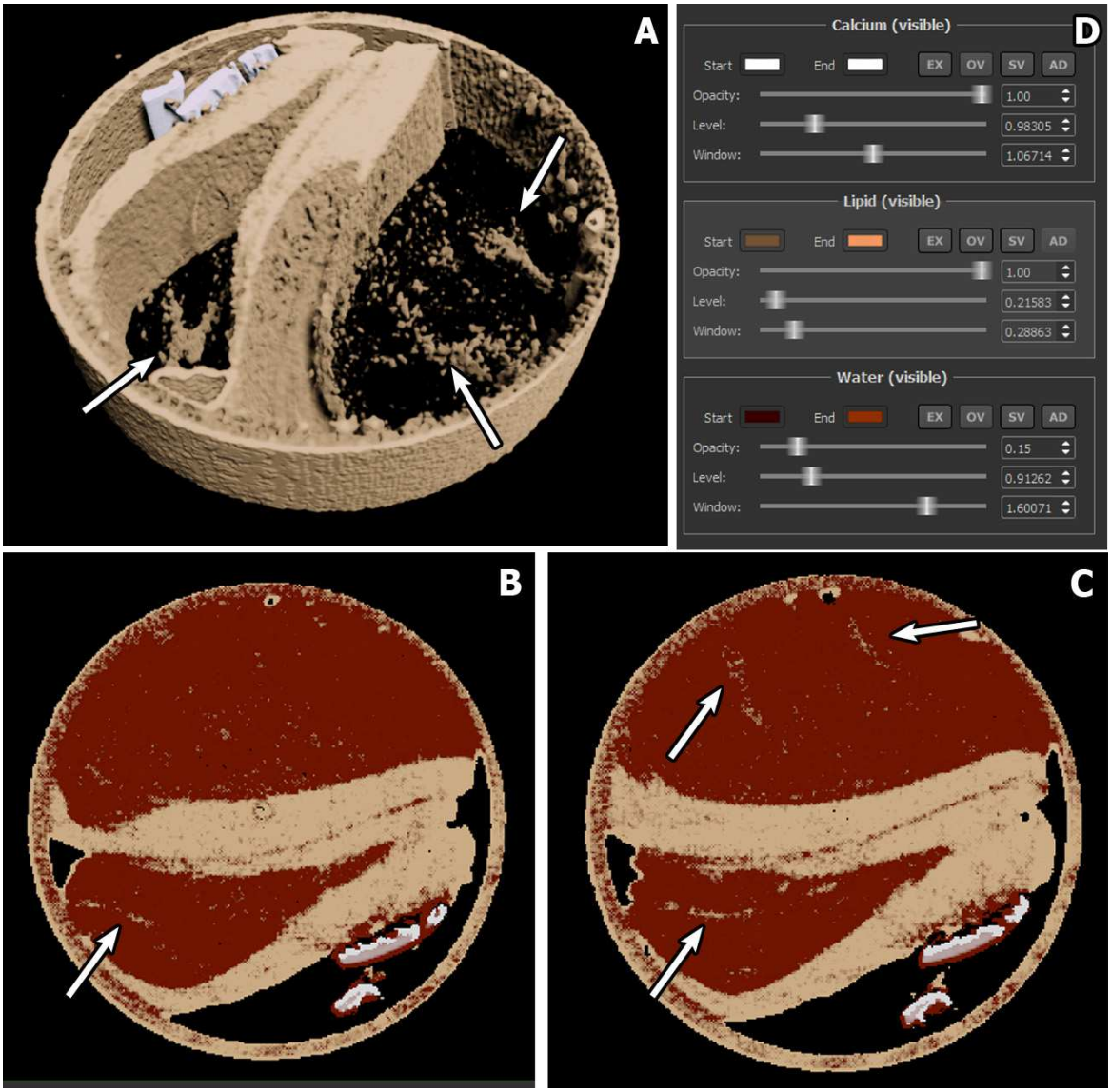


Figure 9.10: Visualisation of marbling patterns (marked by arrows) in the Meat1127 dataset with MARS Vision. **A**: DVR showing the lipid channel (beige) and the calcium channel (white). **B**, **C**: 2D spectral mode visualisation of two slices, with all three materials shown. **D**: the transfer function settings used in **A**-**C**.

Table 9.4: Basic information about the Meat1127 dataset.

Parameter	Description
Name	Meat1127
Year acquired	2013
Object scanned	Piece of lamb meat and bone
Study objective	Discrimination between soft tissue and fat tissue using spectral CT
Energy volumes	4
Material volumes	3 materials (water, fat, calcium)
Data formats	Linear attenuation (energy volumes), relative concentration (material volumes)
Dataset dimensions	$436 \times 436 \times 126$ voxels
Bits per voxel	16

9.4 Cartilage imaging

The MARS project is conducting research into using spectral CT to assess the degradation of knee cartilage, which is usually caused by osteoarthritis [326]. Arthritis is one of the leading causes of disability in countries such as New Zealand [327] and the USA [328].

Osteoarthritis leads to the gradual loss of cartilage in the joints. This loss can be quantified using CT through the use of an ionic iodine-based contrast agent such as Hexabrix [329]. Cartilage usually overlaps with bone, which means that separating these two tissue types during visualisation and analysis is difficult. However, when a contrast agent is used, it only accumulates in the cartilage, which makes tissue discrimination much simpler.

The concentration of Hexabrix (or another iodine-based contrast agent) is inversely related to the concentration of a polysaccharide called glycosaminoglycan (GAG), and can serve as a quantitative marker for evaluating cartilage health [330]. As cartilage deteriorates, its GAG content decreases. This leads to increased absorption of the contrast agent. Therefore, the less iodine is absorbed by the cartilage, the healthier it is.

9.4.1 Knee Cartilage dataset

The Knee Cartilage dataset is a scan is of an excised section of a human tibia containing cartilage tissue that has been treated with Hexabrix. Basic information about this dataset is summarised in Table 9.5. A paper describing the methods of acquiring, processing, and analysing this dataset is currently being prepared for submission to the *European Radiology* journal by a team of MARS researchers, led by K. Rajendran. My contribution to this paper includes a visualisation of the distribution of calcium and iodine, similar to the images shown in this section.

Table 9.5: Basic information about the Knee Cartilage dataset.

Parameter	Description
Name	Knee Cartilage
Year acquired	2014
Object scanned	Piece of knee bone and cartilage treated with an iodine-based contrast agent (Hexabrix)
Study objective	Assessment of knee cartilage health
Energy volumes	4
Material volumes	2 (calcium, iodine)
Data formats	Linear attenuation coefficients (energy volumes), absolute concentration (material volumes)
Dataset dimensions	$544 \times 460 \times 587$ voxels
Bits per voxel	16

This study demonstrates that the MARS system is able to measure the concentrations of both iodine and calcium. As mentioned above, overlapping cartilage and bone are often hard to distinguish, so cartilage itself was not a target material for decomposition; the iodine channel was used instead. These two material channels are shown in Figure 9.11A.

In this case, there are two requirements: the need to remove the sample tube and the MD calibration tubes (already discussed in section 8.5), and the need to clearly show the concentration gradient for the iodine volume. This was done by creating a colour gradient that ranges from blue to red, showing the regions of low and high concentration of iodine, respectively. Calcium is not a material of interest, and is only used to provide the context. Therefore, the precise colour

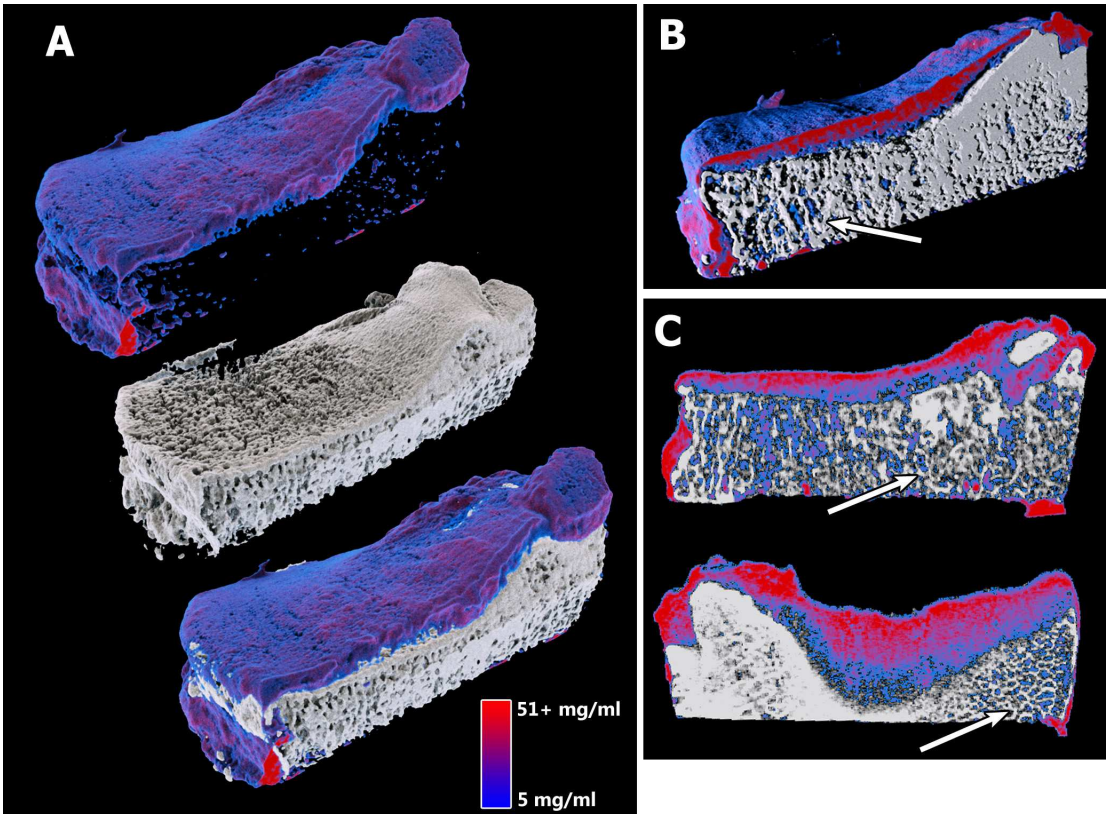


Figure 9.11: **A**: DVR of the iodine (blue/purple/red) and calcium (white) channels of the Knee Cartilage dataset. **B**: DVR with a clipping plane used to reveal the detail inside the dataset. **C**: 2D spectral mode slice visualisation. Some iodine has seeped through the cartilage into the bone (marked by arrows).

gradient applied to this channel was not important, and a plain grey/white colour scheme was chosen.

Gradient visualisation can be carried out using both 2D and 3D techniques, as shown in Figure 9.11B and C. If DVR is used, then clipping can be used to show the gradient at a particular location in the volume (Figure 9.11B). Clipping planes are required because the standard ramp transfer function shape (discussed in section 6.2), leads to occlusion of ROIs, which are the regions of high concentration of iodine. Visualisation of these regions is an important objective of this study, as that is where the cartilage is most damaged. However, they are obscured by the surface of the iodine volume (shown in blue), which corresponds to the areas of low concentration of iodine.

Visibility of ROIs can be improved by using the window transfer function shape. A narrow window can be used to show a small range of concentrations, as illus-

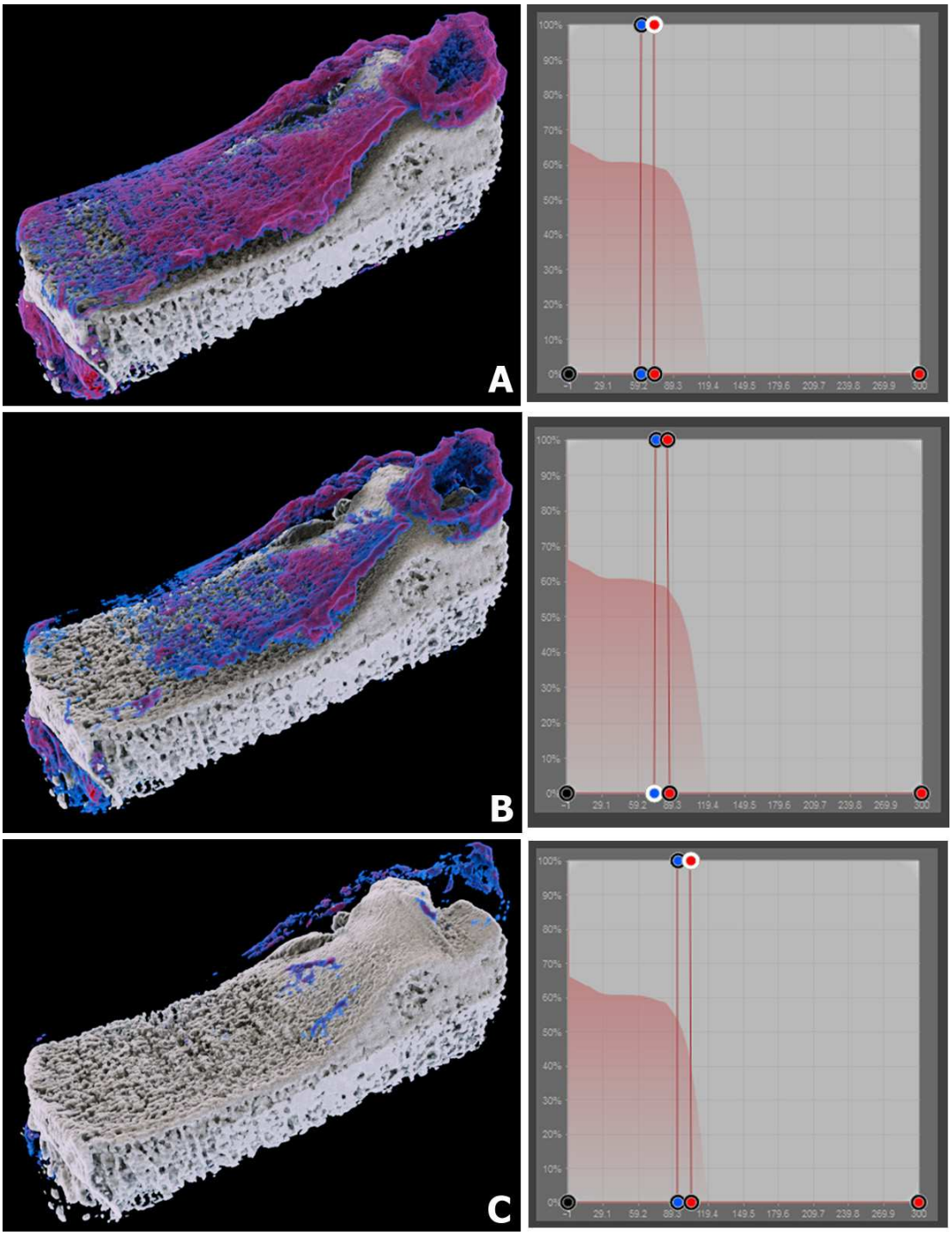


Figure 9.12: Using the window transfer function shape to visualise a narrow range of concentrations of iodine. **A**: 62-73 mg/ml. **B** 73-84 mg/ml. **C**: 93-104 mg/ml. No clipping planes are required. The regions containing these concentrations of iodine are clearly shown. The transfer functions created by the simple transfer function editor are shown on the right.

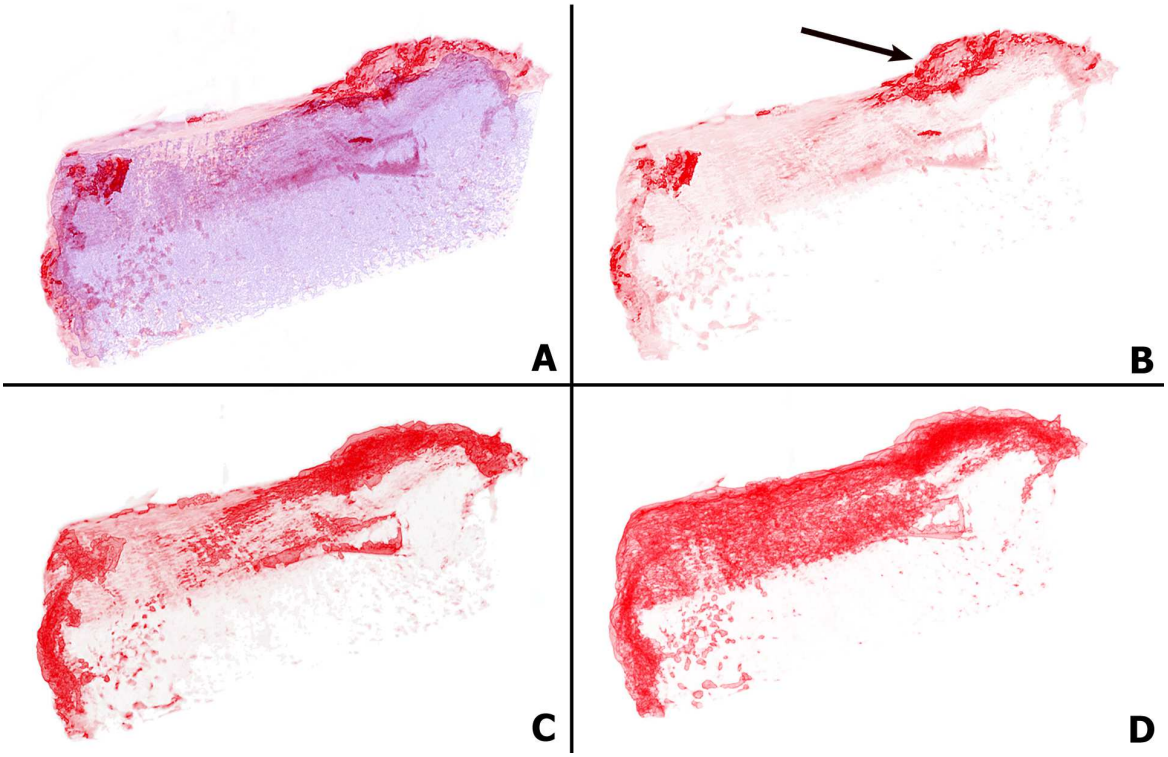


Figure 9.13: TIP visualisation of the Knee Cartilage dataset. **A**: iodine (red) and calcium (purple) channels. The calcium volume provides the context. **B**: iodine channel only, with the threshold set at 100 mg/ml, and a narrow window. **C**: threshold at 80 mg/ml. **D**: threshold at 60 mg/ml. From these images, we can see that the cartilage has the poorest quality at the inner ridge (marked by an arrow), as that is where the concentration of the iodine is highest.

trated in Figure 9.12. The same can be done using the threshold intensity projection with a narrow window, as shown in Figure 9.13. TIP can be used to locate the densest parts of the iodine volume, and allows for a clear visualisation of the areas where cartilage has deteriorated the most.

Finally the Knee Cartilage dataset does not need to be visualised with the help of magic lenses or overlays because both materials are distributed reasonably evenly throughout the dataset, and both occupy a large volume of space. In contrast, some materials in datasets such as Mouse12 (section 9.6.1), Plaque72 (section 9.2.1), or Meat1127 (section 9.5), occupy a large volume of space, while others are localised to a small region of space. In those cases, the context can be provided by one of the large volumes (for example, the water or lipid volumes), while the localised material (for example, calcium), can be shown inside a magic lens.

9.4.2 Summary

Knee cartilage imaging is a recent addition to the set of clinical applications studied by the MARS project. From the point of view of visualisation, it is not a complex problem, as there is only a single material of interest (an iodine-based contrast agent). The context can be provided by the calcium volume, or by a soft tissue volume, if one is created during material decomposition.

The variation in the concentration of iodine can be displayed by creating a transfer function with the simple editor, and using 2D spectral mode or DVR visualisation. The window transfer function shape is particularly useful for displaying a narrow range of concentrations, while hiding the rest of the iodine channel. TIP can also be used for the same purpose.

9.5 Metal implant imaging

This section describes the visualisation of datasets acquired during research into beam hardening artefact reduction with the MARS system. Beam hardening is a common problem in CT imaging, as described in section 3.3. It poses a serious problem, because the attenuation profile of the affected area changes significantly, sometimes by up to 10% [331]. This change severely distorts the image, produces dark streaks behind highly-attenuating objects (such as bones or metal implants), and renders the boundary between these objects and the surrounding tissues invisible.

Beam hardening does not affect narrower energy bins (that is, the bins measuring a small energy range) as much as wider energy bins [332]. The reason is that current CT reconstruction algorithms make poor assumptions about the nature of the x-ray beam, assuming that it is monochromatic (all photons are the same energy), while in reality it is polychromatic (contains a range of photon energies). As discussed in section 2.1.2, photon absorption by a material varies with respect to energy, so an approximation that assumes that all photons are the same will generate image artefacts, which are referred to as beam hardening. Refer to Chapter 3 of *The Essential Physics for Medical Imaging* by Bushberg et al. [20] for more details.

Therefore, a CT system that measures more than one bin is able to compare the differences in attenuation among the wide and the narrow bins and potentially reduce the severity of beam hardening artefacts [333]. The MARS system is able

to partition the measured x-ray spectrum into up to eight energy bins of arbitrary width, and is therefore well-suited for research into beam hardening artefact reduction.

The aim is to establish the settings (such as the energy bin ranges) that lead to effective beam hardening artefact reduction. The knowledge gained from these preliminary studies is likely to result in the creation of beam hardening correction algorithms that can be incorporated into the MARS data processing toolchain.

9.5.1 Titanium screw in bone

This dataset is a scan of a titanium screw inserted into a piece of bone, and consists of four energy volumes, as shown in Table 9.6. It has also been described in the paper by Rajendran et al. [155]. I have contributed to this publication by creating the images that show the interface between bone and titanium, and the reduction in beam hardening that can be achieved using the MARS system. Similar images are shown in this section.

Table 9.6: Basic information about the Titanium Screw in Bone (TiScrew) dataset.

Parameter	Description
Name	TiScrew
Year acquired	2014
Object scanned	Titanium screw inserted into a piece of bone
Study objective	Reduction of beam hardening artefacts using spectral CT, visualisation of the boundary between bone and metal
Energy volumes	4 (15-120, 35-120, 60-120 and 80-120 keV)
Data format	Linear attenuation coefficients
Dataset dimensions	618x618x128 voxels
Bits per voxel	16

The aim of this study was to visualise the difference in the interface between the bone and the metal, and to measure the improvement achieved by using narrow energy bin ranges.

Beam hardening can be visualised using 2D or 3D techniques, as shown in Figure 9.14. If DVR is used, then the transfer functions for energy volumes must be

created manually. However, the transfer function is not complex, as only two materials need to be classified. The attenuation of bone and titanium is substantially different, and the attenuation ranges do not overlap in this dataset.

This makes the transfer function design straightforward, as shown in Figure 9.14D. The metal and bone regions need to appear visually distinct. This is the only requirement, so simple colour-coding is sufficient. Therefore, after consulting with K. Rajendran, the primary investigator of this study, I have decided to present bone in grey, metal in red, and make the air regions appear transparent.

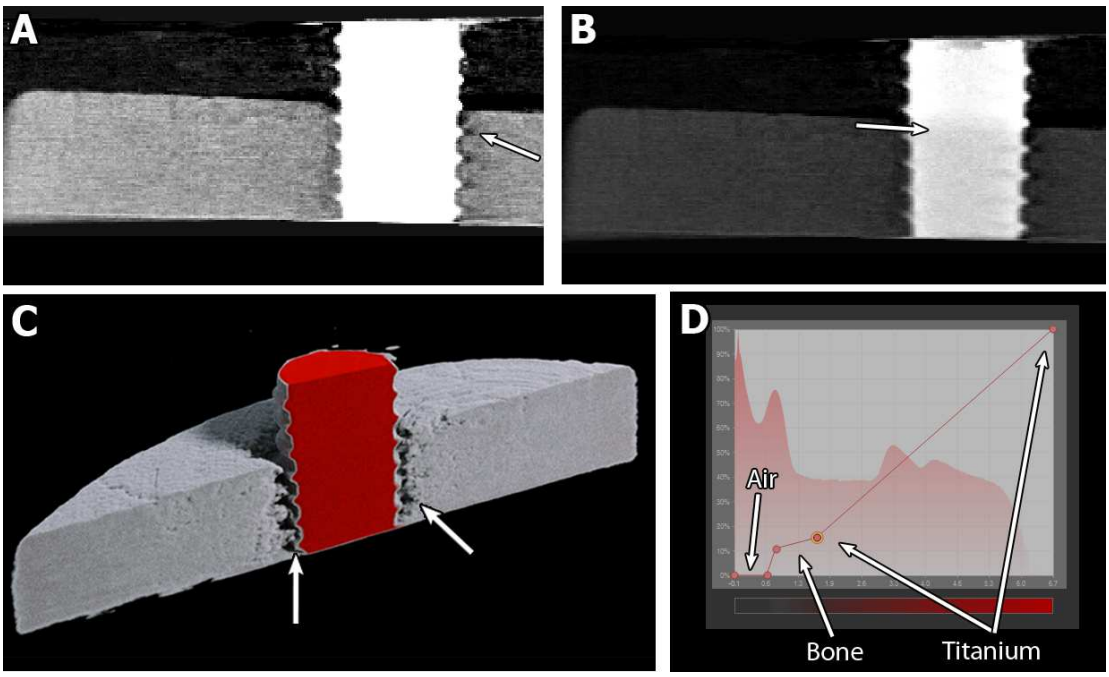


Figure 9.14: Visualisation of beam hardening in the TiScrew dataset. **A**: dark areas at the boundary between bone and metal (marked by an arrow). **B**: cupping effect, where the attenuation of the metal inside the screw appears to be different than the attenuation near the surface. **C**: DVR of the 15-120 keV volume, with the clipping plane set to show the same slice as **B**. The transfer function has been designed to show the difference between air, bone, and metal. The visible gaps between the screw and the bone (marked by arrows) are a result of beam hardening, which has reduced the linear attenuation coefficient of bone in these regions to the same value as air. **D**: the transfer function used to generate the image in **C**.

Another problem is the large amount of noise present in the 80-120 keV energy volume, as shown in Figure 9.15A. During visualisation, this noise, originally caused by a smaller number of available photons (due to the width of the energy bin), stands out as speckles inside regions of air. It can be effectively suppressed by using cubic B-spline interpolation, as shown in Figure 9.15B.

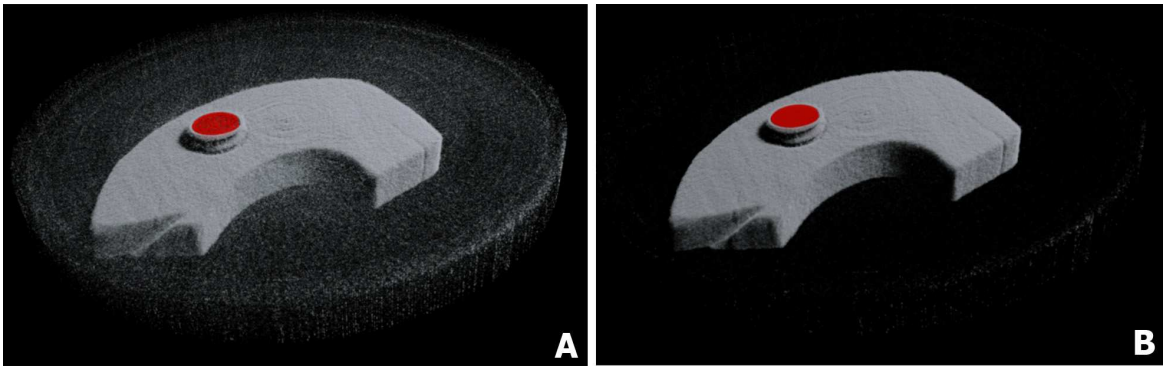


Figure 9.15: **A**: the 80-120 keV energy volume of the TiScrew dataset visualised with the default MARS Vision DVR settings. **B**: cubic B-spline interpolation used instead of the default trilinear interpolation. The speckle noise is greatly reduced.

The advantage of this technique is that it is purely visual: the source data is unchanged and all measurements are unaffected. This is an important requirement because there are no firmly established techniques for processing MARS spectral CT datasets. Therefore, it is currently unknown how much data processing (including denoising) is required, and which methods should be used. At this point in time, this decision is left to the researchers working with the MARS system; more formal procedures are likely to be established in the future.

Measurement is an important aspect of metal implant imaging, as beam hardening reduction must be quantified, as well as visualised. MARS Vision's tool collection is well-suited for this task. Two tools can be used for this purpose:

- The profile of a line tool. This tool measures the linear attenuation or concentration at certain intervals along a line drawn by the user (section 8.1.2). The measurements are presented in a graph, as shown in Figure 9.16E. Alternatively, measurements can be exported for further analysis with external software.

  MARS Vision extends this standard tool to simultaneously generate and display the profile of a line in multiple volumes. Datasets such as TiScrew are an excellent example of why such tools are necessary: the differences in the attenuation profile can be compared on the same graph.

- The standard ROI statistics tool. For example, a region on the boundary between bone and metal can be selected and measured. Another option is to

measure the attenuation of an ROI inside the screw. As before, measurement of all volumes can be conducted simultaneously. The use of this tool can be better illustrated using the CoCr ball dataset, described in the next section.

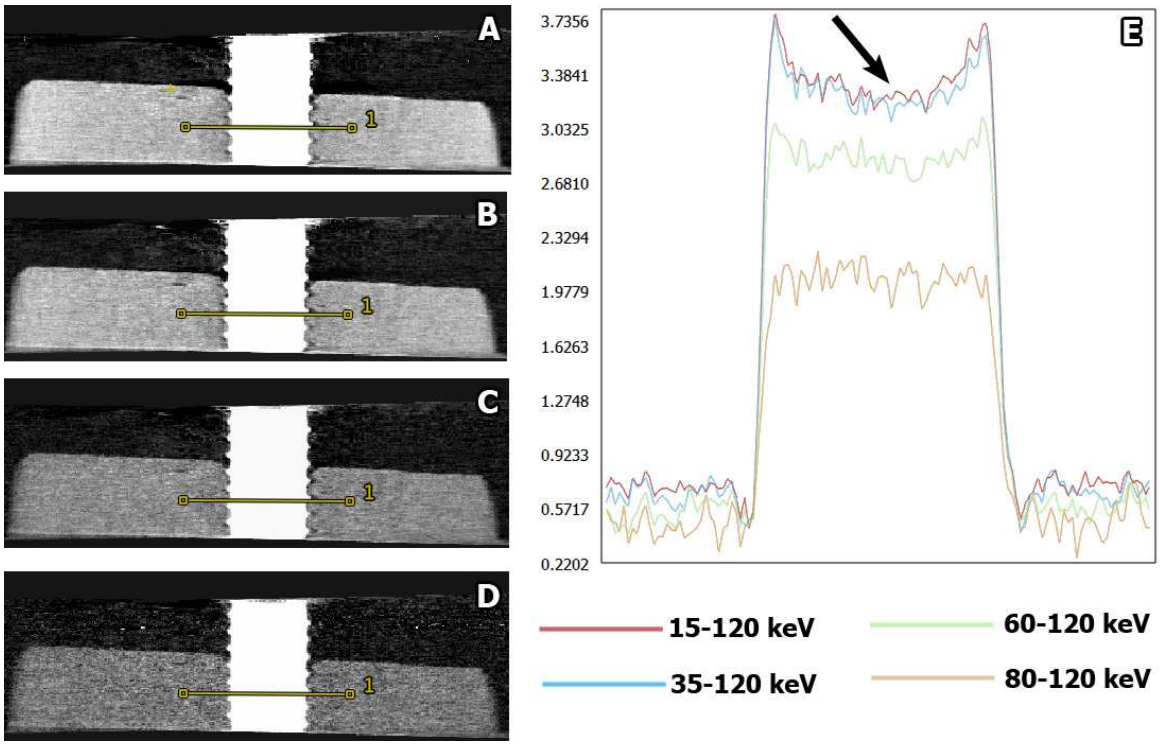


Figure 9.16: **A**-**D**: visualisation of a slice from four energy volumes of the TiScrew dataset. The volumes are: 15-120, 35-120, 60-120 and 80-120 keV, respectively. **E**: a graph of the profile of a line drawn across these images, generated by MARS Vision. The first two energy volumes are clearly affected by beam hardening (the visible dip in the line profile graph, marked by an arrow), while the 80-120 keV energy volume is largely unaffected.

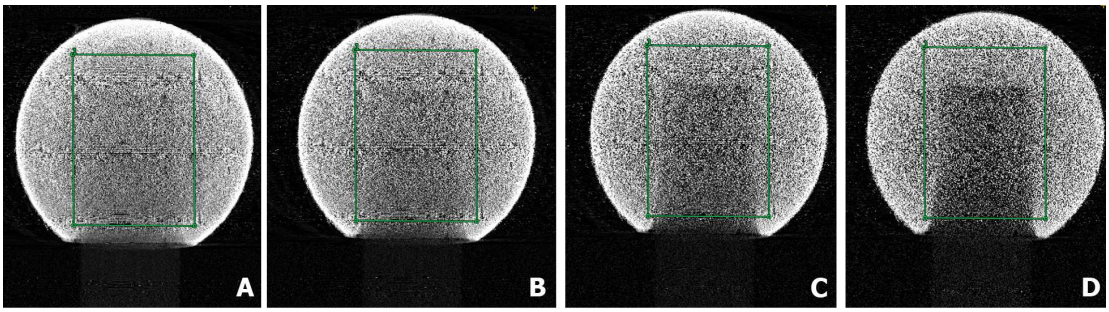
9.5.2 CoCr Ball

This dataset is a scan of a hip joint implant made of a cobalt and chromium (CoCr) alloy, which is commonly used for surgical implants due to its high resistance to corrosion [334]. The implant is a hollow sphere, mounted on a PMMA holder. The aim of this study was to evaluate the potential of using spectral CT to reduce the cupping artefacts caused by beam hardening. Information about this dataset is summarised in Table 9.7.

Table 9.7: Basic information about the CoCr Ball dataset.

Parameter	Description
Name	CoCr Ball
Year acquired	2013
Object scanned	A hip joint implant made of a cobalt and chromium alloy, mounted on a cylindrical PMMA sample holder
Study objective	Beam hardening artefact reduction using spectral CT
Energy volumes	4 (50-120, 60-120, 70-120 and 80-120 keV)
Data format	Linear attenuation coefficients
Dataset dimensions	436x436x126 voxels
Bits per voxel	16

The cupping effect has already been displayed in Figure 9.16E. The graphs for the wider energy bins (for example, 50-120 keV) show a clear "dip" in the middle of the ball. This reduction of the measured attenuation coefficient is called "cupping". The cupping effect inside the CoCr Ball dataset has already been visualised in section 8.4, and a profile of a line graph has been shown in section 8.1.2. The difference can also be studied by designating an ROI and measuring all four volumes, as shown in Figure 9.17.



Volume	Slice	Type	Min	Max	Average	Median	Mode	Std. dev.	No. voxels
50-120 keV	250	Polygon	-0.1	7.12409	1.79543	1.76113	1.76113	0.944584	27006
60-120 keV	250	Polygon	-0.1	8.48966	1.68496	1.60603	1.60603	1.11491	27006
70-120 keV	250	Polygon	-0.1	9.60684	1.57777	1.45094	-0.1	1.35499	27006
80-120 keV	250	Polygon	-0.1	8.53146	1.40778	1.14075	-0.1	1.38784	27006

E

Figure 9.17: **A**-**D**: slices from four energy volumes of the CoCr Ball dataset, with the ROI marked in green. **E**: statistics for the selected ROI.

9.5.3 TiMesh

The last dataset discussed in this section is the TiMesh, which is a small conical vase made from titanium. Information about this dataset is summarised in Table 9.8. TiMesh has been described in the paper on beam hardening artefact reduction by Rajendran et al. [9]. I have contributed to this publication by carrying out all data processing needed to convert the energy volumes into a format suitable for visualisation, designing the transfer functions and all other 3D visualisation parameters, and generating images that show the effects of beam hardening.

Table 9.8: Basic information about the Titanium Mesh (TiMesh) dataset.

Parameter	Description
Name	TiMesh
Year acquired	2013
Object scanned	Small (around 5 cm in height) titanium scaffold in the shape of a vase
Study objective	Metal artefact reduction using the MARS system
Energy volumes	4 (15-80, 35-80, 60-80 and 60-80 keV)
Data format	Linear attenuation coefficients
Dataset dimensions	$436 \times 436 \times 126$ voxels
Bits per voxel	16

Visualisation of the TiMesh dataset is very similar to the TiScrew and CoCr Ball datasets. The first task that needs to be performed is ring artefact removal, because, as described in section 3.3, one end of the dataset contains severe ring artefacts. This portion can be removed through the use of clipping planes, as shown in Figure 9.18. The majority of the dataset is unaffected, and can still be visualised and measured.

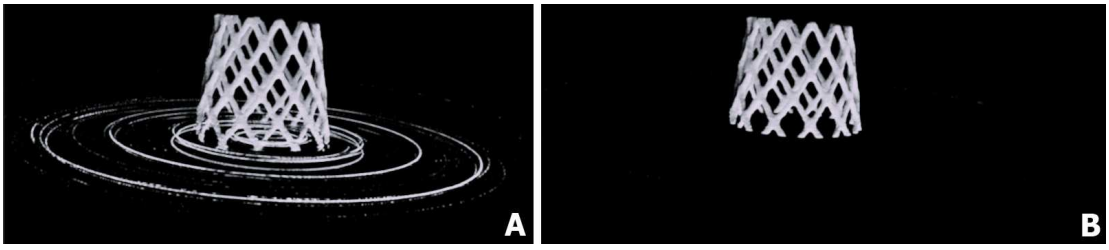


Figure 9.18: **A**: the entire TiMesh dataset. **B**: ring artefacts removed through the use of a clipping plane.

The beam hardening artefacts can be visualised using traditional 2D techniques, or with DVR, as shown in Figure 9.19. A similar image was used in the paper that I have co-authored with Rajendran et al. When DVR is used, the visualisation must show both the air region, and the titanium scaffold. This can be accomplished by:

1. Rendering the titanium mesh as a solid grey surface.

2. Rendering the air as a translucent green cloud. This allows for a clear visualisation of the streaks that are caused by beam hardening (which distorts the measured attenuation of air and makes the air region appear inhomogeneous).

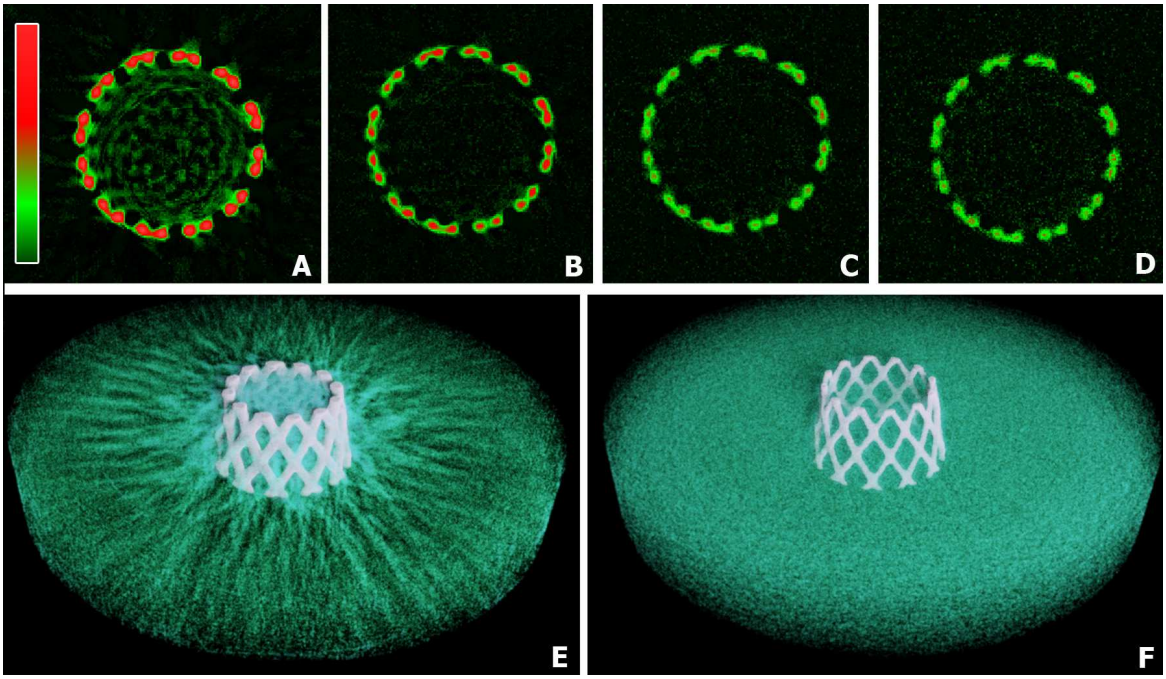


Figure 9.19: **A**-**D**: 2D visualisation of a slice from four energy volumes of the TiMesh dataset. A look-up table is used to assign colour. The energy volumes are: 15-80, 35-80, 60-80 and 60-80 keV, respectively. **E**: DVR of the 15-80 keV energy volume. Clearly visible streaks are emanating from the titanium scaffold. **F**: DVR of the 60-80 keV energy volume. The air is shown as a much more homogeneous region.

Finally, measurements can be performed using the profile of a line tool, or using the ROI tool. The use of these tools has already been demonstrated in earlier sections.

9.5.4 Summary

Visualisation of metal implant datasets is different from the visualisation of other types of samples scanned by the MARS system. This is because material decomposition is not typically performed, and reconstructed energy volumes are visualised and measured directly.

From a technical perspective, the visualisation of the boundary between a metal implant and bone, air, or soft tissue, requires the use of MARS Vision's advanced transfer function editor. However, the transfer function does not need to be complex, as it only needs to separate the bone from the surrounding tissues.

This imaging application requires measurement tools to compare the attenuation profiles of ROIs across different energy volumes. This can be done by directly comparing the statistics calculated for the same ROI in multiple energy volumes, or by comparing the profiles of lines drawn through these ROIs. Other tools, such as magic lenses, may also be used, but are usually not required due to the simple geometry of the scanned objects.

9.6 Small animal imaging

The final spectral CT imaging application described in this chapter concerns small animal studies conducted by the MARS team. Such research is essential for evaluating the capabilities of the MARS system, as small animals (usually mice) are the closest human analogue that can fit inside the MARS scanner. For other work on contrast agent detection in small animals using spectral CT refer to Cormode et al. [5] and Pan et al. [224, 225].

Mouse models are used for assessing the spread and distribution of contrast agents, or the compounds tested during drug development [335]. Contrast agents are usually injected into mice and allowed to naturally circulate and spread over a certain time period (for example, 24 or 48 hours). The mouse is then either sacrificed and scanned, or anaesthetised and scanned alive.

Research indicates that the visualisation and measurement of the spread of contrast agents is an important application of spectral CT [2, 5, 10, 55]. However, these substances are usually located inside, or next to soft tissues, which makes occlusion during 3D visualisation almost inevitable. Therefore, currently, the visualisation of mouse datasets is a challenging technical problem.

The MARS project has carried out several studies that involved scanning mice or other small animals [6, 13, 53, 57, 227]. I have contributed to the paper by Walsh

et al. [13], for which I have visualised the gold nanoparticles (AuNP) injected into a mouse. I have used MARS Vision to generate an image that shows the distribution of gold inside the kidneys, along with the context, which was provided by the bones, as shown in Figure 9.20. The advanced transfer function editor had to be used for this purpose, as MD was not available.

Figure 9.20: DVR of the mouse dataset acquired, processed and visualised by Walsh et al. [13]. The kidneys are visible because they contain gold nanoparticles that have accumulated inside. However, the separation of kidneys and bones must be done manually, as this dataset only contains energy volumes.

9.6.1 Mouse12

This dataset is one of the oldest spectral CT datasets acquired by the MARS team. It has originally been acquired by Anderson et al. [6] to determine whether spectral CT is capable of discriminating between multiple radiological contrast agents injected into a mouse. In addition to being presented in the original paper, Mouse12 has become one of the standard datasets used for testing the visualisation tools

and algorithms developed by the MARS project [74, 129]. Table 9.9 summarises the basic information about this dataset.

Table 9.9: Basic information about the Mouse12 dataset.

Parameter	Description
Year acquired	2010
Animal scanned	Mouse
Study objective	Discrimination between different types of contrast agents injected into a mouse.
Energy volumes	4 (15-80, 23-80, 30-80, and 35-80 keV)
Material volumes	Originally none, 3 materials (calcium, barium, iodine) have been extracted from the original energy volumes in 2014
Data formats	Linear attenuation coefficients (energy volumes), relative concentration (material volumes)
Dataset dimensions	$440 \times 338 \times 257$ voxels
Bits per voxel	16

The original paper by Anderson et al. described the use of PCA for discriminating between the three materials of interest: calcium, iodine and barium. The advances in MARS material decomposition algorithms have led to the extraction of the same materials using an entirely different technique (basis material decomposition [36]).

For visualisation, the 15-80 keV energy volume has been combined with the three extracted materials. This makes an excellent test case for the 2D and 3D material overlay technique, as shown in Figure 9.21. The ROI, containing two contrast agents (iodine and barium), is located inside the chest of the mouse. The energy volume is used to provide a familiar anatomical context, and the material information can be displayed alongside it. This technique is currently also used for dual-energy CT and PET-CT data visualisation.

Overlaying a material onto an energy volume distorts the colour gradient assigned to it due to the blending between the colours. This can be seen in Figure 9.21A and C. This is the reason for implementing the colour replacement mode, described in section 5.6.4.1. Colour replacement allows for a clear visualisation of the gradient for the chosen material channel (barium), as shown in Figure 9.21B.

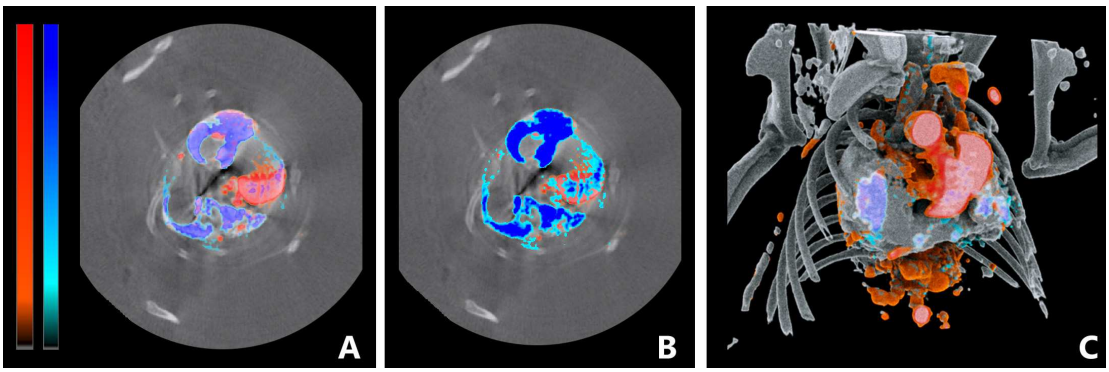


Figure 9.21: Visualisation of the 15-80 keV energy volume of the Mouse12 dataset and the iodine and barium materials. **A**: 2D spectral mode blending. **B**: 2D spectral mode visualisation, with the colour replacement mode used to preserve the gradient assigned to the barium channel. **C**: DVR with a clipping plane set to show the distribution of iodine inside the heart and of barium inside the lungs.

In earlier examples, the use of an energy volume as a basis provided context to the iodine and barium channels. In Figure 9.22, the context is instead provided by the calcium channel, as bones are a clear visual reference. Figure 9.22A shows the use of basic DVR, while Figure 9.22B shows the use of a magic lens to display the ROI without occlusion. Note that the calcium channel is using a "window" transfer function shape, described in section 6.2, which allows a small data range to be shown as a translucent isosurface.

A similar transfer function is used for the energy volume (Figure 9.22E), but the opacity is set to be low. This shows a small energy range that corresponds to soft tissue as a translucent cloud, which provides an outline of the mouse's body. This general technique can also be applied to any other dataset to display the context in a way that does not interfere with the regions or materials of interest.

In conclusion, this dataset does not present a difficult visualisation problem, as the distribution of both materials of interest is restricted to a small area inside the chest. Occlusion can be minimised through the use of magic lenses or TIP, which can be used to show the outlines of materials (shown in Figures 7.10 and 7.12).

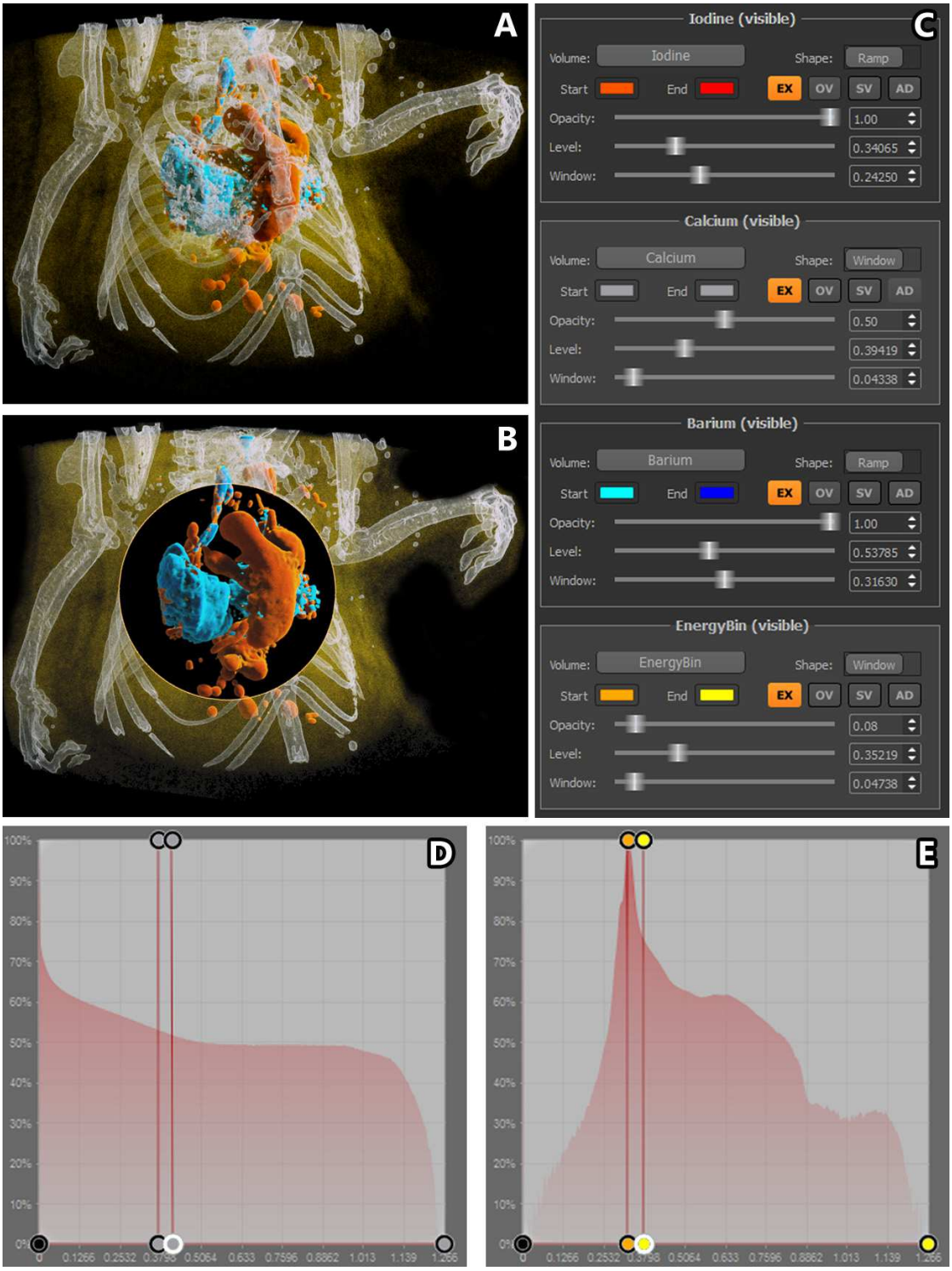


Figure 9.22: **A**: visualisation of the 15-80 keV energy volume of the Mouse12 dataset (yellow) and the three identified materials. **B**: use of the 2D magic lens to reduce occlusion inside the ROI. **C**: the settings of simple transfer function editor for the four channels. **D**: the transfer function graph for calcium. **E**: the transfer function for the energy volume. Note that these transfer functions are not the standard ramp shape, but are instead using the window shape.

9.6.2 Mouse0

The Mouse0 dataset is a recent MARS spectral CT dataset acquired during the research into using AuNP as a spectral CT contrast agent. Information about this dataset is summarised in Table 9.10.

Table 9.10: Basic information about the Mouse0 dataset.

Parameter	Description
Year acquired	2015
Object scanned	A mouse with a tumour implanted under the skin, injected with AuNP
Study objective	AuNP detection in a small animal model using the MARS system
Energy volumes	5 (8.0- 27.9, 27.9-48.2, 48.2-68.1, 68.1-81.2, 81.2-120 keV)
Material volumes	4 (water, lipid, calcium, gold)
Data formats	Linear attenuation coefficients (energy volumes), absolute concentration (material volumes)
Dataset dimensions	$583 \times 583 \times 99$ voxels
Bits per voxel	16

This dataset can be used to clearly illustrate the advantages of 2D visualisation. Occlusion, as often mentioned in this thesis, is a very serious problem during 3D visualisation. The effects of occlusion are particularly severe for small animal datasets, as mouse anatomy is much more complex than the structure of any other object currently scanned by the MARS system.

Figure 9.23 shows the synchronisation of 2D and 3D visualisation. DVR (A) suffers from occlusion, but can still be used to find the approximate locations of ROIs. Once found, an ROI can be marked with an annotation; 2D views can then be centred on this point. Finally, various forms of 2D slice visualisation (B-E) can be used to study the dataset's features in greater detail.

Visualisation of AuNP in the Mouse0 dataset provides a good example of the utility of the magic lens exclusion mode (section 7.4.2.2). In this situation, the contrast agent has spread throughout the body of the mouse, while the ROI (the tumour implanted under the skin) is relatively small. The gold injected into the mouse has accumulated in the tumour, but has also spread into other organs, as shown in Figure 9.24A.

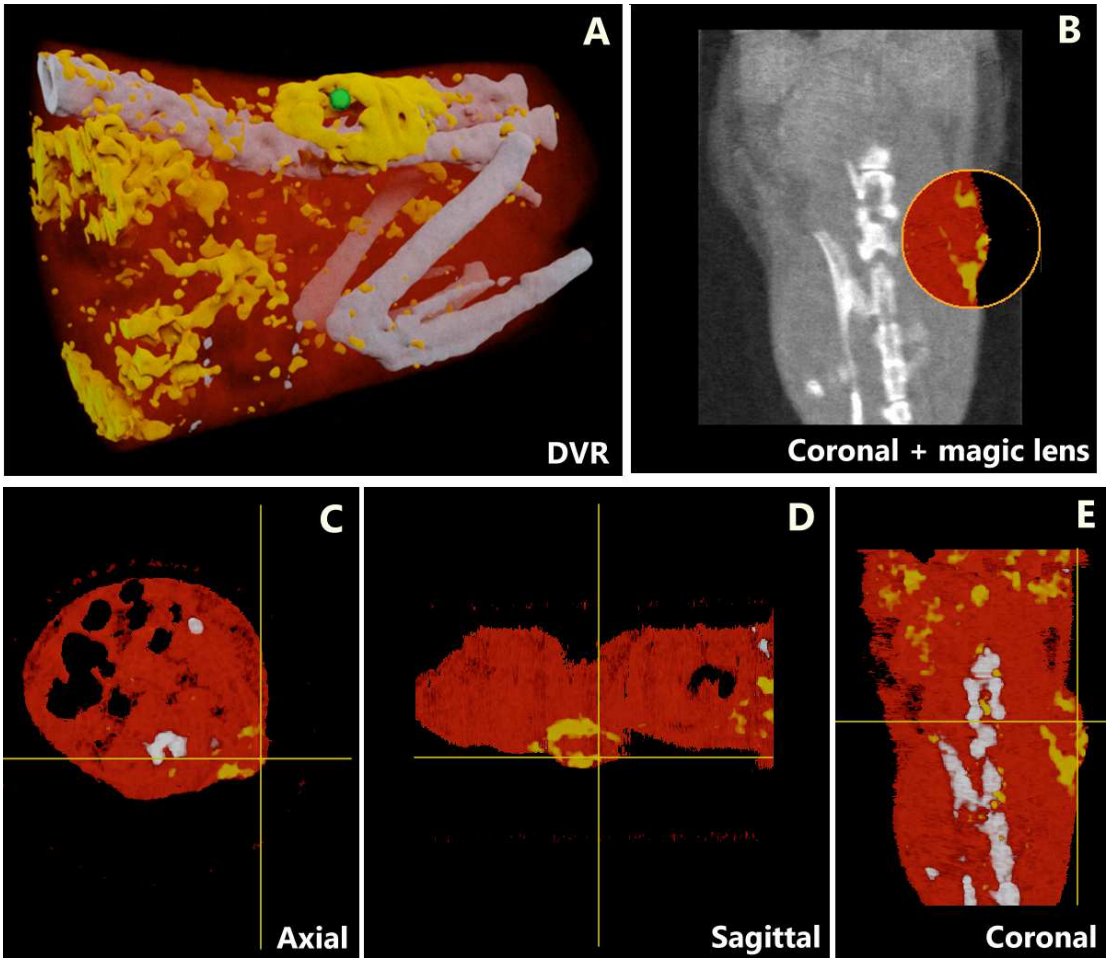


Figure 9.23: Synchronisation of 2D and 3D views in MARS Vision. The location of the 3D cursor, used to synchronise the visualisation, is rendered as a green sphere (painted using 3D annotations, as described in section 8.1) in **A**, and as yellow crosshairs in the slice views shown in **C**-**E**. The colour schemes applied to soft tissues and bone follow the convention used for visualising atherosclerotic plaque and lamb meat datasets (described in sections 9.2 and 9.3, respectively).

The concentration of gold in the tumour and in the other organs is very similar, so the transfer function cannot be adjusted to hide the gold located outside of the ROI. Volume editing tools (section 8.5) can be used to crop out all gold outside of the ROI, but this is a destructive process that leads to the loss of information.

Therefore, the only option for *temporarily* removing the unwanted structures highlighted by gold is to use a magic lens. Using the magic lens in the standard mode (Figure 9.24B) is ineffective: the ROI is displayed clearly, but the rest of the gold is not removed. However, using the magic lens in exclusion mode substantially

reduces the visual clutter, as shown in Figure 9.24B. Therefore, this mode can be used for restricting the visualisation of particular material or materials to a certain ROI,

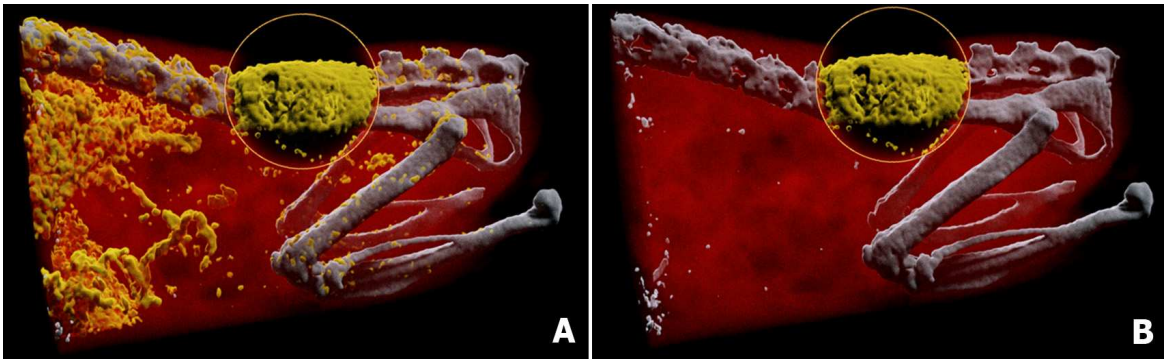


Figure 9.24: Using the magic lens exclusion mode to reduce the number of features displayed in a scene. **A**: standard 2D magic lens. **B**: 2D magic lens with the exclusion mode enabled.

9.6.3 Summary

Small animal datasets present a difficult visualisation challenge due to the complex internal anatomy of the specimen. A combination of the volume data editing, magic lens, and overlay tools, may be used to clearly display the location of the ROIs. Assigning colour gradients to materials of interest with the simple transfer function editor is usually sufficient. Energy volumes can be visualised with the help of the advanced transfer function editor; however, this is usually not required, as the materials of interest have already been identified and separated. Finally, 2D and 3D view synchronisation is a good option for avoiding the negative effects of occlusion: the ROI can be found using DVR, and then viewed without occlusion using 2D slices.

9.7 Other datasets

The performance and image quality of DVR algorithms is commonly assessed using standard, freely-available volumetric datasets. Previous work on spectral CT data visualisation has also used such datasets to test rendering algorithms [74, 129].

Evaluating the capabilities of MARS Vision using conventional CT datasets is of limited value because its GUI and tools have been designed specifically for

working with spectral CT data. Nevertheless, we can identify the The Visible Human Male dataset as being particularly useful for testing, as it is a full-body single-energy CT dataset that can be used to approximate future human-scale spectral CT datasets. The MARS project does not have any similar datasets that may be used to test whether the tools developed over the course of this research are applicable to human imaging. Therefore, the only option is to simulate a human-scale spectral CT dataset.

Other datasets, such as the Carp [77], have also been used in several sections to illustrate certain concepts, or show the use of tools. However, they will not be discussed here, as all visualisation techniques potentially applicable to human spectral CT imaging can be demonstrated using the Visible Human Male dataset.

9.7.1 Visible Human Male dataset

This is a well-known single-energy CT dataset acquired in 1995 as part of the Visible Human project initiated by the National Library of Medicine [162]. It was acquired by scanning the corpse of a 38-year old male (a prisoner executed by lethal injection), who willed his body to the Texas State Anatomical Board. This dataset has been widely used for research and education [336, 337, 338], and can also be used to simulate the visualisation of future human-scale MARS spectral CT datasets.

To simulate such a dataset, at least one material needs to be identified. MARS MD algorithms cannot be used for this purpose, as they are specifically designed for working with multiple energy volumes (section 3.2.2). Instead, we may use MARS Vision's region growing segmentation tools (section 8.3) to extract a portion of the lungs and treat it as a material volume. This process creates a dataset comprising two volumes: the original energy volume, and the faux "contrast agent" material volume. Basic information about the modified Visible Human Male dataset is provided in Table 9.11.

Since this dataset consists of two volumes, the overlay, magic lens, and TIP tools can now be used. For example, TIP can be used to set the thresholds for the context and the material of interest, and render the two volumes simultaneously, as shown in Figure 9.25.

Figure 9.26 shows that the visualisation techniques used for currently-available MARS small animal datasets may also be applicable to human datasets. The same

Table 9.11: Basic information about the Visible Human Male dataset.

Parameter	Description
Name	Visible Human Male
Year acquired	1995
Object scanned	Human male prisoner executed by lethal injection.
Study objective	Research into human anatomy.
Energy volumes	1
Material volumes	1 (simulated "contrast agent" volume, created by segmenting the energy volume with MARS Vision)
Data formats	Linear attenuation coefficients (energy volume), absolute concentration (material volume)
Dataset dimensions	$512 \times 512 \times 746$ voxels
Bits per voxel	16

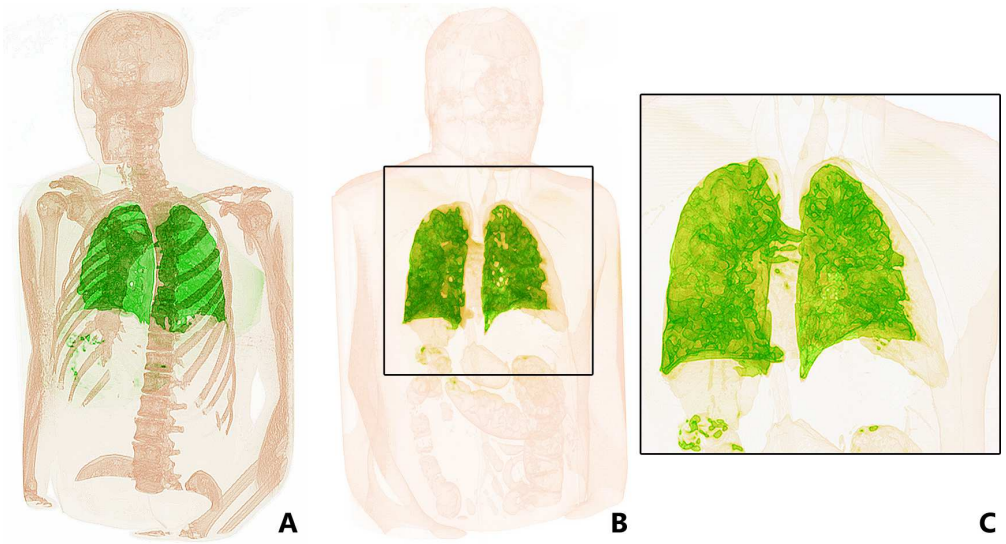


Figure 9.25: **A**, **B**: TIP rendering of the Visible Human Male dataset with the extracted "contrast agent" volume shown in green, using different thresholds for both channels.**C**: close-up of the ROI inside **B**.

approach, consisting of synchronised 2D and 3D visualisation, and overlays, magic lenses, and TIP for displaying ROIs, yields similar results.

The colour gradient for the contrast agent can be designed using the simple transfer function editor (for example, using the ramp, or the solid gradient shapes). The advanced transfer function editor can be used to classify the soft tissues and

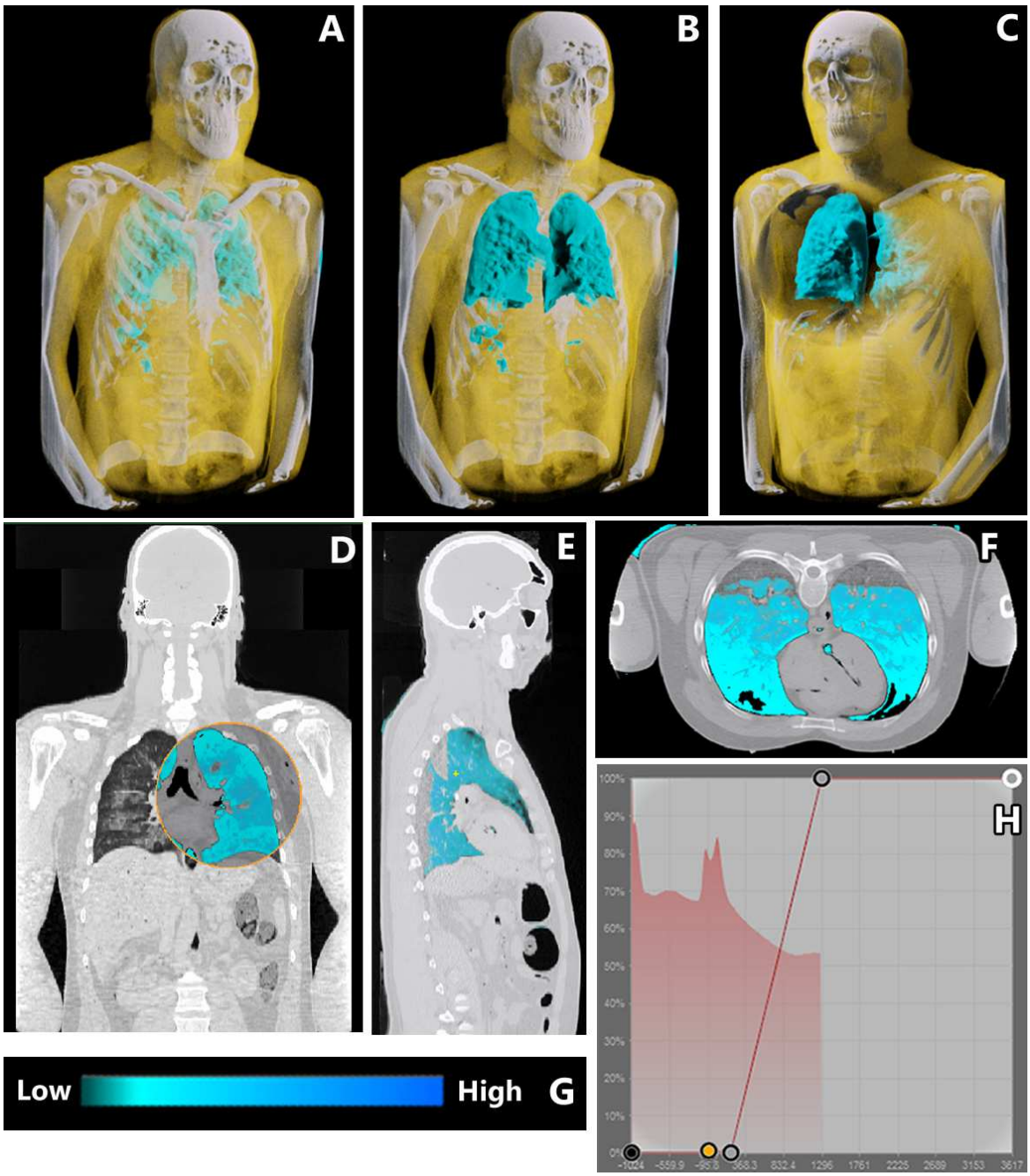


Figure 9.26: **A**: DVR of the Visible Human Male dataset with the extracted "contrast agent" volume shown in blue. **B**: overlay by the contrast agent. **C**: 3D magic lens. **D**: 2D slice visualisation using the magic lens. **E**, **F**: 2D spectral mode visualisation. **G**: the colour gradient created for the contrast agent by the simple transfer function editor. **H**: the transfer function for the energy volume, created by the advanced editor.

the bones for 3D rendering (Figure 9.26A, B, C, and H). The classification does not need to be precise, as the energy volume is only used as context. Measurements can be made in the 2D view, in which case the traditional greyscale colour scheme, controlled by window and level parameters, can be set for the energy volume (Figure 9.26D, E, F).

9.7.2 Summary

Several conventional CT and MRI datasets have been used for testing MARS Vision. Of these, the only notable one is the Visible Human Male dataset, which provides a way of simulating the visualisation of a full-body spectral CT dataset. Testing using this dataset shows that all techniques currently applied to MARS datasets can also be translated to human-scale spectral CT imaging.

9.8 Summary

This chapter has demonstrated the use of a combination of tools for achieving effective visualisation of currently-available MARS spectral CT datasets. The range of clinical applications discussed in this chapter covers most of the currently-known applications of spectral CT technology.

The visualisation requirements of each imaging problem have been discussed, and the use of MARS Vision's tool collection has been demonstrated. In each case, MARS Vision is able to display the structure of the dataset and clearly show the location of ROIs. Once the ROIs have been found, analysis can be performed using standard (statistics inside the ROI), or extended measurement (CVH or profile of multiple lines) tools.

Finally, this chapter has discussed the application of these tools and techniques to future human spectral CT datasets. In particular, the overlay mode, threshold intensity projection and magic lens tools are likely to be useful for displaying the ROIs inside human-scale body spectral CT datasets.

Chapter X

Discussion: the future of spectral CT visualisation

This chapter combines the discussion of the work presented in this thesis with the description of the future work in the area of spectral CT data visualisation. This thesis has examined the requirements for effective visualisation of currently-available MARS spectral CT datasets, and developed solutions for the majority of known problems. However, continued development of spectral CT technology opens numerous possibilities for further research. All aspects of visualisation, from volume rendering algorithms to transfer function editors and presets will likely be studied in greater detail in the future.

This chapter is structured as follows. Section 10.1 discusses the results obtained over the course of this research, identifying the advantages and limitations of the proposed solutions. Section 10.2 discusses the future work in the area of spectral CT visualisation and interaction. It covers a number of topics, such as:

- The likely changes resulting from the introduction of spectral CT into clinical practice.
- The future evolution of GUIs of spectral CT data visualisation software.
- The deficiencies of current rendering techniques and the means for improving them.
- The development of interaction techniques, and, in particular, the beginning of research into multi-modal interaction with spectral CT data visualisation.

This chapter concludes with a summary of the scientific contributions made over the course of my PhD research.

10.1 Discussion of results

This thesis serves as the first in-depth study of spectral CT data visualisation. Due to the paucity of research in this field, the fundamental requirements needed to be established first. This required examining the nature of spectral CT data, analysing the tools used previously, consulting with the pre-clinical researchers from the MARS team, and reviewing the present and future applications of spectral CT imaging. This section summarises the observations made over the course of this research.

10.1.1 Tools for visualising MARS datasets

Due to the diversity of the applications of spectral CT and the complexity of many imaging problems, no technique is sufficient on its own. Instead, a combination of several specialised tools must be used to solve the current visualisation problems.

This research project has created such a toolset, which consists of:

- Various 2D and 3D visualisation techniques (Chapter 5), which support the fusion of data from an arbitrarily large number of channels.
- A specialised transfer function editor (Chapter 6) for designing transfer functions for both energy, and material volumes.
- The tools for minimising occlusion (Chapter 7).
- Measurement tools extended to take advantage of spectral CT data (section 8.1).
- The tools for interactively editing volume data (sections 8.3, 8.4, and 8.5)

This is the key reason for the adoption of MARS Vision by the MARS team. There is no known comparable software application described in literature, except for MARSCTExplorer, the software used prior to MARS Vision. MARSCTExplorer was solely focused on 3D visualisation, while the complexity of many imaging applications required a more diverse toolset.

For example, the magic lens tool is a potential solution when occlusion of ROIs poses a serious problem, as it does during small animal or atherosclerotic plaque

imaging. On the other hand, metal implants and scaffolds are structurally simple objects, and the ROIs (the boundaries between metal and another material) are usually not occluded. This means that basic 2D or 3D visualisation (without additional tools such as magic lenses) is sufficient.

Chapter 9 has demonstrated that the tools included in MARS Vision are sufficient for the visualisation and analysis of currently-available MARS spectral CT datasets. These tools meet the study goals of the MARS pre-clinical research teams, and have been used to display and analyse data acquired during research into applications of spectral CT imaging [2, 4, 9, 11, 13, 155].

Finally, testing using a simulated full-body spectral CT dataset suggests that these tools may also be applicable to future human-scale spectral CT datasets. This remains an area for future research, as discussed in section 10.2.2.

10.1.2 Advantages and limitations of material visualisation

As Chapter 6 has shown, designing transfer functions for material volumes is significantly easier than creating transfer functions for energy volumes. A simple colour gradient can be created by selecting a pre-defined transfer function shape, and setting several parameters (window, level, and two colours, as explained in section 6.2). The colour gradient can then be directly mapped to a concentration gradient.

However, at present, material volumes do not capture the full anatomy, or structure, of the scanned object. MARS MD is incapable of identifying individual organs or tissues; instead, the decomposition is limited to the air-like, water-like, and lipid-like components, as well as chemical elements, such as calcium, iodine or gold [36]. Visualising each material separately, or combining multiple material channels, does not provide the same amount of anatomical information as directly visualising energy volumes.

This is the reason for the popularity of the technique of overlaying a material onto an energy volume. It has been used for spectral CT and dual-energy CT data visualisation, as described in sections 4.2 and 4.3. Other modalities, such as PET-CT and fMRI, do not generate material information, but use the same approach to display the functional information alongside the anatomical information, as noted in section 4.1.

The advantage of this hybrid approach lies in providing the users with a data

format they are likely to already be comfortable with, and enhancing the visualisation by supplying additional functional or material information. In the case of spectral CT, the energy volumes contain the necessary anatomical context, while the material volumes provide the information about the composition of tissues.

The advantages of energy and material data fusion have been demonstrated by visualising contrast agents in small animal models, as described in section 9.6. Mice and other small animals are the closest organism to a human that the MARS system is able to scan, and pose a much more challenging visualisation problem than other types of objects, such as excised tissue samples or phantoms. The reason is that the anatomy is complex, and the spread of contrast agents inside the body is not completely predictable (as opposed to the distribution of a material in a specially prepared phantom).

Therefore, the exact location of ROIs in mouse (or human) datasets is not necessarily known in advance. ROIs can be found by visualising one or more materials without any context, but, usually, this is not the case. As demonstrated in section 9.6, an energy volume, used as the context, can be easily fused with material volumes without the need to design complex colour schemes.

In the future, the need for an energy volume to provide the context may be eliminated by the introduction of tissue classification algorithms, capable of separating individual organs or tissue types. This possibility is discussed in section 10.2.1.

10.1.3 Transfer functions for spectral CT data visualisation

Chapter 6 of this thesis discussed the properties of material volumes and proposed a novel transfer function editor for working with material volumes. This editor, described in detail in section 6.3, creates a 1D transfer function for colour and a 1D transfer function for opacity. The two transfer functions are controlled by window and level parameters.

This direction was taken because the pre-clinical researchers working for the MARS team were familiar with the concept of window and level adjustment during 2D slice visualisation. The proposed transfer function editor uses the same concept, requires little configuration, leverages the users' existing knowledge, and allows them to create transfer functions for both 2D slice visualisation, and 3D DVR. As demonstrated in Chapter 9, it is able to generate transfer functions suitable for

visualising a wide range of MARS spectral CT datasets.

Another possible direction is the use of multi-dimensional transfer functions, which have proven to be effective for visualising both medical, and scientific datasets [266, 267, 268]. In particular, they can improve material identification by analysing the behaviour at the boundaries between materials. This is highly useful during energy volume visualisation.

However, the MARS toolchain already includes material decomposition, which is carried out as an automatic pre-processing step. When visualising MARS datasets, the users only need to assign colour and opacity to several material volumes, but do not need to manually extract any materials from energy volumes. Therefore, one of the major reasons for the use of multi-dimensional transfer functions does not apply to MARS datasets. In addition, MARS users are unfamiliar with multi-dimensional transfer functions, or even transfer functions in general. The editor proposed in Chapter 6 hides the process of transfer function design from the users and displays a simple graphical representation of the colour scheme assigned to each volume.

Nevertheless, multi-dimensional transfer functions for spectral CT data need to be investigated in the future. Separation of materials during visualisation of energy volumes is no longer required; instead, multi-dimensional transfer functions may be used to extract organs and tissues from multiple material volumes. This would be a complex task for inexperienced users, but pre-set transfer functions may be provided to simplify the workflow.

In conclusion, this thesis is the first work that studied the question of transfer function design for MARS spectral CT data. As such, it focused on simpler solutions that extended the concepts known to MARS users. It demonstrated that, at this time, multi-dimensional transfer functions are not required for visualising spectral CT data. However, based on prior research, there is reason to think that they may improve the quality of visualisation and convey more information to users. Therefore, multi-dimensional transfer functions, as well as the editors for creating them, remain an interesting and promising future research direction.

10.1.4 Volume data editing tools

A portion of this thesis has been dedicated to assessing the impact of noise and image artefacts on the visualisation of MARS datasets, and on creating the meth-

ods for removing them. This is a common problem for an experimental imaging system such as MARS.

Currently, MARS data processing algorithms are unable to eliminate a number of artefacts, including speckle noise, streaks and rings. In addition, structures such as sample tubes and calibration tubes are not removed automatically, and frequently occlude the ROIs. Therefore, at this stage, manual volume data editing is a necessary task, along with the removal of certain occluding structures, such as calibration tubes and sample tubes (section 8.5). These actions, strictly speaking, are not visualisation tasks, but they are required in order to clearly display the ROIs inside many MARS datasets.

In the future, this problem will likely be solved by the improvements to the detector hardware used in the MARS scanner, and to the software algorithms in the MARS data processing toolchain. For example, work is currently being carried out to automate the calibration of MARS MD algorithms to avoid having to attach calibration tubes to each scanned sample.

This thesis has also addressed some aspects of the problem of segmenting spectral CT datasets and subsequent visualisation of segmented data. In particular, threshold-based whole volume and region growing segmentation (section 8.3) has been implemented. Segmented volumes can be combined with the overlay, magic lens and threshold intensity projection tools, which have been described in Chapter 7. However, the development of segmentation algorithms specific to spectral CT data has not been an aim of this research, and only the most basic approaches have been discussed. While even basic segmentation has been demonstrated to be useful in certain circumstances, further development of segmentation algorithms may make it easier for the users to identify and extract ROIs.

10.1.5 Limitations of using colour

The visualisation techniques described in this thesis rely on the use of colour to distinguish between the channels of a spectral CT dataset. This type of image fusion is similar to existing medical imaging modalities, such as PET-CT (described in section 4.1.1), with one important difference: the number of channels that may be need to be fused is much greater. Currently, the MARS system acquires up to different eight energy volumes, and identifies up to seven materials [36]. However, the number of distinct colours is limited. This means that, at some stage, the number

of materials identified by spectral CT MD algorithms may exceed the number of colours that humans can easily distinguish, especially when the materials overlap, and the colours are blended.

At this point, it is difficult to judge whether this problem will limit the number of channels displayed simultaneously. Pre-clinical studies have been limited to identifying no more than 4-5 materials, which have usually not been shown in a single image (section 4.3). Generally, a material has either been shown separately, or overlaid onto some form of context. In that case, a colour (or a colour gradient) has been used to show the material of interest, while the context has been presented using a greyscale colour scheme.

The observations made during this research suggest that simultaneous visualisation is best limited to around three or four channels. The reason for this is the distortion of the colour gradient assigned to each channel by blending with another colour, as discussed in section 5.6.4. However, this number can increase if the materials do not overlap, or if no colour gradients are used (that is, if each material is associated with a single colour).

If multiple gradients need to be visualised, then one channel can be set as the context, which is usually shown in greyscale, or using another similarly desaturated colour scheme. This leaves a large colour space for creating distinct gradients for two or three materials of interest. Alternatively, the colour replacement mode (sections 5.3.5.1 and 5.6.4.1) can be used to preserve the colour gradient for a single channel.

The currently-known applications of spectral CT mostly target one or two materials of interest (for example, gold, iodine, or calcium) in a single scan. Therefore, colour coding is unlikely to become a limiting factor, unless further research determines an application where the simultaneous visualisation of a large number of channels is necessary. This issue will likely need to be re-visited when spectral CT imaging develops further.

10.1.6 Performance and dataset size

Currently-available MARS spectral CT datasets are not large enough to pose a problem for the visualisation techniques discussed in this thesis. Usually, the size is under 1 GB, and, as section 5.5 has demonstrated, all MARS datasets can be rendered at interactive frame rates. In most cases, a low- or medium-quality image

containing all visible structures will be displayed in under a second, while a high-quality rendering suitable for inclusion in a publication may take several seconds to render.

However, as the MARS project moves towards building a human-scale spectral CT scanner, the DVR algorithm will almost certainly need to be re-implemented. The problem of scaling to larger dataset sizes (for example, of 100 GB or more) has not been considered, and it is unlikely that interactive visualisation of such datasets at full resolution will be possible. There are two reasons for this:

1. For DVR, all volumes must be fully loaded into GPU memory, and each volume must be stored in a single contiguous array. The GPUs currently used to render MARS datasets usually contain around 4-6 GB of video random access memory (VRAM). This amount is sufficient for all current datasets.

   However, the increase in the size of MARS datasets is almost certain to outpace the increase in the amount of GPU VRAM. The proposed full-body MARS scanner will likely retain the spatial resolution of the existing small specimen model. This means that the voxel size will be around $100\,\mu m^3$, and that $1\,mm^3$ will contain 1000 voxels. A single volume of a full-body dataset will therefore contain around a trillion voxels ($700\,mm \times 700\,mm \times 1700\,mm \times 1000 = 833,000,000,000$ voxels), with 2 bytes required to store each voxel. This is at least 1000 times larger than the number of voxels in a single volume of the largest currently-available datasets. In comparison, the largest amount of VRAM on a modern GPU is 12 GB (section 2.2.1).

2. The step size used by MARS Vision's DVR is adjusted based on the dimensions of the dataset, as described in section 5.3.3. Usually, a ray advances by a distance roughly equal to the size of 1 voxel (though this distance is configurable). For example, if an object was scanned with a voxel size of $1\,mm^3$ and $100\,\mu m^3$, DVR would be around 10 times slower in the second case. Therefore, even if the dataset can fit inside a GPU's VRAM, the performance of DVR may be non-interactive.

To solve this problem, level of detail rendering techniques will need to be investigated. These are discussed in section 10.2.3.

On the other hand, the algorithms for 2D slice visualisation (window/level adjustment and spectral mode blending, described in section 5.6) scale well. Currently, due to the small size of slices, performance is not an issue, and a basic CPU implementation is sufficient. However, these algorithms, are, in fact, embarrassingly parallel, as each image pixel is processed independently. In addition, the amount of data required to render a single slice is relatively small. Therefore, these algorithms are well-suited for GPU implementation.

For example, a $700\,mm \times 700\,mm$ axial slice of a volume of a hypothetical full-body MARS dataset can already be rendered using existing CPU and GPU hardware. If the voxel size is $100\,\mu m^3$, then each slice will contain 7000×7000 pixels. While large, this size is similar to the size of slices of current micro-CT datasets, or photographs taken by high-resolution digital cameras. In conclusion, when MARS datasets grow in size, 2D slice visualisation algorithms will simply need to be re-implemented in CUDA, but will not require major changes.

10.1.7 Applicability to other spectral CT systems

The work described in this thesis has been carried out as part of the MARS project, and MARS spectral CT datasets have been used to test the visualisation algorithms and data processing tools. Therefore, the applicability of these tools to the data produced by other spectral CT systems needs to be discussed.

These tools have been specifically designed to be as generic as possible, so that they may be applied to most spectral CT datasets. For example, the threshold intensity projection, overlays, or magic lenses, can be used with any multi-volume dataset, as these techniques do not rely on any particular property of MARS energy or material data, or spectral CT data in general. Likewise, the hybrid transfer function editor is a GUI element that can be used to design transfer functions for any volumetric dataset, and the fusion performed by MARS Vision's DVR and 2D spectral mode algorithms is not specific to any data type.

The segmentation and volume data processing tools, as well as the methods of reducing noise and removing artefacts, have been designed specifically to fit into the MARS toolchain and address some of the deficiencies of the existing data processing, reconstruction and MD algorithms. However, they do not depend on any particular data format. For example, the 2D polygon crop and 3D voxel removal tools (section 8.5), which are essential for many MARS datasets (as the

scanned objects are usually enclosed in a sample tube), can also be used to remove occluding structures in any volumetric dataset.

This is supported by the results of the testing conducted using conventional CT datasets. In particular, the Visible Human Male dataset, described in section 9.7.1, is a good example: MARS Vision has been demonstrated to be able to load a standard DICOM series created by another CT system and use a number of tools to simulate the visualisation and analysis of a human-scale spectral CT dataset. The Carp CT dataset, used to demonstrate the segmentation techniques implemented in MARS Vision, is another example (section 8.3).

This process of editing the Visible Human Male dataset involved segmenting out the lungs (to simulate a contrast agent identified by MD), assigning a colour gradient to the faux "contrast agent" volume, and rendering two volumes simultaneously. The magic lens and TIP tools were used to show the location of the ROI, while 2D spectral mode slice visualisation was used to show the distribution of the "contrast agent" inside the body.

If another spectral CT system produces a similar dataset, then the 2D and 3D data fusion algorithms, as well as the overlay, magic lens, and TIP tools, could be applied to it in a similar manner. The dataset may need to be converted to a format supported by MARS Vision, but the *process* of visualisation and analysis would be unaffected.

In conclusion, the only aspects of this research that are not applicable to other spectral CT systems concern the integration of MARS Vision into the MARS software toolchain.

10.1.8 Standardisation

Currently, spectral CT data formats are not standardised, which limits the options for sharing data between different research teams, and developing universally-applicable tools and presets. The MARS team is using the DICOM file format to store energy and material data; however, all energy volumes are stored as frames inside a single DICOM series (section 5.2.3), while each material volume is stored in a separate DICOM series.

This decision has been made by the MARS development team due to the limitations of the current DICOM standard for conventional CT data, as described in section 5.2. Therefore, in order to load a complete MARS spectral CT dataset,

the viewing software must be able to load multiple DICOM series and treat them as a single multi-variate dataset. However, this is not a universal feature of DICOM viewers. This is not a problem for the MARS team, but, for example, it may restrict other researchers who may want to study MARS datasets, some of which have already been published [4, 9]. These datasets are Meat1127, described in section 9.3.1, and TiMesh, described in section 9.5.3.

The lack of standardisation is only a problem due to the novelty of spectral CT technology. DICOM standards for energy and material data produced by spectral CT will certainly be introduced as this technology gains widespread acceptance.

10.1.9 Summary

Spectral CT is currently in the early stages of development, which leads to a number of issues, such as the lack of standards, the experimental hardware and software that cause image artefacts, and the paucity of datasets useful for testing. In addition, until now, the process of visualising spectral CT data (including MARS data) has remained a poorly-explored area, as little research into the visualisation methods, tools, and techniques has been conducted.

This research has addressed these problems by establishing the requirements for effective visualisation of currently-available MARS spectral CT datasets, creating a suite of custom visualisation and analysis tools, and demonstrating that they can be used for clearly displaying the ROIs inside MARS datasets. Some limitations, such as the reliance on colour to discriminate between the different channels, and the inability to visualise extremely large datasets, remain. However, it is expected that most of these problems will be solved in the future, as discussed in the next section.

10.2 Future work

The field of spectral CT data visualisation presents numerous opportunities for further research. Every aspect of visualisation and interaction that has been explored by this research project can be examined in greater detail. This section discusses these possibilities.

10.2.1 Spectral CT data processing and tissue classification

As shown in this thesis, the processing applied to spectral CT data determines what users see when they visualise the data, how they interact with the visualisation, and what measurements they can perform. In particular, the introduction of MD has significantly changed the way datasets were studied, as direct assessment of the composition of the scanned object became possible. Therefore, it is important to consider how future advances in spectral CT data processing may influence the visualisation process.

The MARS project has already fully integrated material decomposition into its data processing toolchain. The next step may involve classifying voxels based on the types of tissue they are likely to belong to, as opposed to the current approach of looking at specific materials, such as water, calcium, or iodine. For instance, a full-body spectral CT dataset could be classified into several tissues, such as muscle, fat, bones, liver, brain, blood vessels. Each contrast agent will likely be stored as a separate volume, similar to the way MARS material volumes are currently stored. Following the convention used in this thesis, we may refer to these new datasets as "tissue volumes", or "tissue type volumes".

At this time, it is unknown which techniques will be used for visualising and interacting with tissue volumes. However, these techniques may not be substantially different from those currently used for working with material volumes. The reason is that both the hypothetical tissue classification, and the existing material decomposition, partition the dataset in a very similar manner.

MD identifies materials and separates them into volumetric datasets, where a voxel represents the concentration of a material. Tissue classification is likely to be very similar, and will also separate tissues or organs into independent volumetric datasets. Therefore, the tissue and material data types are much closer to each other than they are to energy volumes, which nearly always contain several materials and/or tissues at once.

When working with energy volumes, the user must manually separate these materials or tissues. This leads to a specific set of requirements, which usually includes a complex but powerful transfer function editor (section 6.1). On the other hand, when no manual separation of materials (or tissues) is required, then the user interfaces for creating colour schemes can be simplified, as discussed in section 6.1.1.

Further, since tissue classification algorithms will almost certainly produce multiple volumes, the overlay, magic lens, and TIP tools should be useful for visualising occluded regions of interest. For example, a magic lens may be used to view a volume of liver tissue while preserving the context, which can be provided by the other tissue type volumes.

Finally, in the future, the creation of energy volumes may be bypassed entirely. The MARS group is working on developing the algorithm for simultaneous material reconstruction (SMR) which combines reconstruction and material decomposition into a single step, by directly reconstructing material volumes from projection data. SMR uses an accurate physical model that eliminates problems such as noise and beam hardening. As a result, the accuracy of material decomposition should be greatly improved.

If this technique replaces standard reconstruction and MD, then the workflow and GUIs presented in this thesis will need to be reassessed: the composition of a spectral CT dataset will change, and all techniques that involve working with energy volumes will no longer apply. This will present new challenges, such as the need to provide an alternative form of anatomical context during material visualisation. Currently, this context is frequently provided by energy volumes, which means that their removal will seriously affect the way users interact with the visualisation of spectral CT data.

10.2.2 Introduction of spectral CT into clinical practice

Section 4.4.7 has argued that the current state of spectral CT technology, as well as the composition of the user base, do not allow for the development of a standard medical GUI for spectral CT data visualisation. However, the situation will change once spectral CT enters clinical use.

This will likely happen in the next few years, subject to further advances in detector design and data processing algorithms [2, 3]. Assuming it takes place, it will result in a substantial change in the composition of the user base, adding large numbers of radiologists to the small group of researchers who use spectral CT today.

Radiologists are highly trained in the use of standard DICOM viewers, and are used to a particular workflow for each imaging modality (such as CT, PET, SPECT, MRI, and so on). The changes to this workflow that are caused by the

additional material information provided by spectral CT will need to be assessed. After this assessment, the tools for visualising spectral CT data will need to be refined and adjusted to suit the preferences of clinical users. Therefore, future research in this area will almost certainly involve consultation with radiologists and other medical professionals.

The introduction of spectral CT into clinical practice will likely involve changes to all steps of the data processing toolchain, and will certainly affect the GUIs of visualisation applications and the tools for interacting with datasets. The tools proposed in this thesis are designed primarily for scientific researchers, and the GUI of MARS Vision is different from the GUI of a typical DICOM viewer. The scientific workflow includes manual data processing, for which a large number of tools is required, while the clinical workflow includes performing a limited range of tasks. However, the tools for performing these tasks must be well-designed, reliable, and easy to use [79, 81, 238].

While the influence of this transition on the visualisation process and tools is hard to predict, we can nevertheless identify several likely outcomes:

- Volume data processing tools (section 8.4) and interactive volume editing functionality (section 8.5) will be retained, but not included in all spectral CT data visualisation software. For example, these tools will likely be present in the software used by the technicians who prepare spectral CT datasets for diagnosis [339], but will be absent in the software used by radiologists for diagnosis.

- Data fusion will likely be based on standard colour presets for different materials, as well as on window/level settings. These concepts are commonly used for visualising conventional CT datasets, and this thesis has shown how they may be applied to spectral CT (for designing the transfer functions for material volumes). The simple transfer function editor implemented in MARS Vision (section 6.2) can perhaps be considered an early prototype of a future clinical GUI for spectral CT data fusion.

- A DICOM standard for spectral CT data will be introduced, standardising the way energy and material volumes are stored.

- 3D visualisation is likely to play a peripheral role, since it is not commonly

used for medical diagnosis. 3D techniques are popular for presenting the results of scientific studies, but, in medical practice, 2D visualisation is still prevalent.

I have considered this issue while designing the visualisation tools for currently-available spectral CT datasets. It is the reason why particular attention has been paid to the design of 2D visualisation and data fusion algorithms (section 5.6).

In conclusion, the overall trend in GUI and tool design is likely to be towards the implementation of a standard medical DICOM viewer GUI with simple data fusion tools. 3D visualisation will be retained, but will not be used often.

10.2.3 Rendering techniques

This thesis has shown that modifying an existing DVR algorithm to render multi-volume spectral CT datasets produces high-quality images that approach photo-realistic image quality. Considering the medical visualisation techniques in use today, it is unlikely that better image quality will be required for the purposes of medical diagnosis.

Therefore, increasing the performance of DVR is the primary concern. High-quality DVR can already be carried out on consumer-grade GPU hardware at interactive frame rates, as shown in section 5.5. However, the rendering speed is slow compared to alternative techniques, such as mesh visualisation.

The second concern is the increasing size of datasets, as spectral CT imaging moves to human scale. For example, if the MARS system uses the same voxel size (around $50 - 100\,\mu m$) for full-body CT datasets, then each energy or material volume will contain over a terabyte of data, as discussed in section 10.1.6. Currently-available GPUs contain no more than 12 GB of VRAM, so GPU-based rendering of such datasets at full resolution will not be possible. Therefore, the implementation of level-of-detail rendering techniques (decomposing the volume into blocks of different resolution, and rendering directly from this representation [245, 340]) will likely be required. For example, the data inside the ROI will be rendered at full resolution, while the surrounding tissues will be downsampled to a low-quality approximation to provide the anatomical context.

Finally, improving depth perception is another area for further research. This can be done by attempting to improve existing visualisation algorithms (usually

by incorporating more realistic illumination), or by implementing stereoscopic rendering.

10.2.3.1 Stereoscopic 3D rendering

Stereoscopy is a valuable technique for conveying depth information. Standard volume rendering mostly relies on motion parallax, perspective, occlusion, and illumination to create the illusion of depth. Stereoscopic 3D rendering enhances this illusion by providing a different image to each eye (stereopsis). This adds more depth cues to the scene without interfering with the depth cues already provided by the other methods.

Stereoscopic visualisation may be useful for displaying spectral CT datasets, because they often contain a large number of overlapping features. In particular, the distribution of contrast agents in the mouse datasets described in section 9.6 may be easier to visualise. This remains an area for further research, perhaps including a user study, as mentioned in section 10.2.6.

Stereoscopic 3D rendering has already been integrated into MARS Vision by other members of the MARS team, as shown in Figure 10.1. It has been positively received by current users, who noted the improvement in depth perception during TIP, MinIP or MIP visualisation. These techniques provide fewer depth cues than DVR, as discussed in section 7.2. The use of stereoscopic rendering enhances depth perception, while retaining the distinguishing features of these projections (removal of occluding structures). However, the feedback has been informal, and the utility of this rendering technique should be properly evaluated.

10.2.3.2 Mesh rendering

Mesh extraction and visualisation has been mentioned as a possible approach to visualising spectral CT datasets in section 2.2 due to being substantially faster than volume rendering. The work by other members of the MARS team (in particular, by Harish Mandalika), has resulted in the addition of an experimental mesh extraction and rendering mode to MARS Vision. A mesh is extracted from each visible volume in the dataset using the Marching Cubes algorithm [93], with the user being able to adjust the thresholds. An example is shown in Figure 10.2.

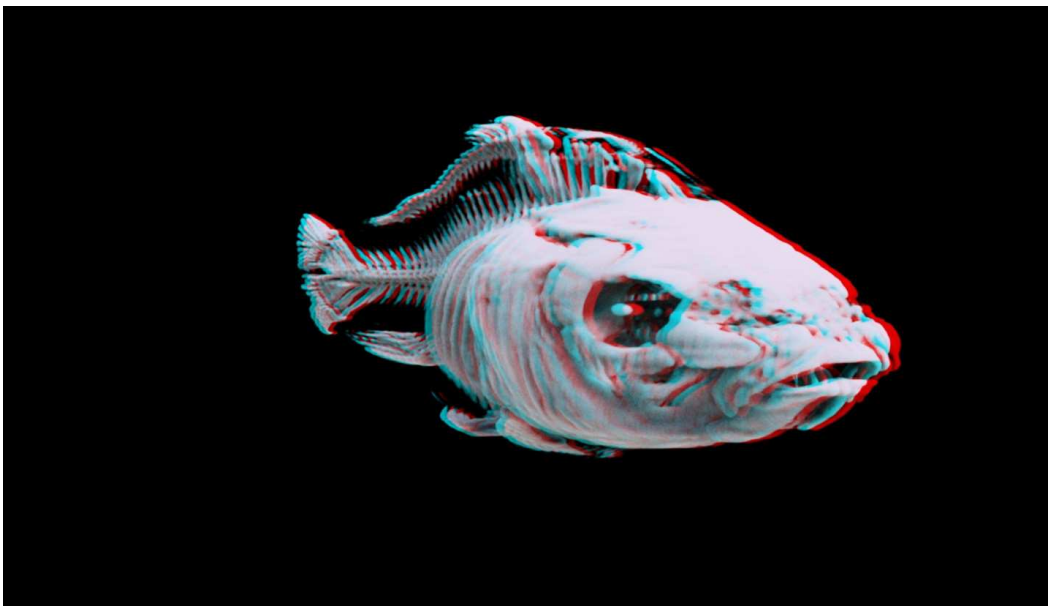


Figure 10.1: Stereoscopic (anaglyph) 3D visualisation of the Carp dataset. Red/cyan glasses are required to view this image correctly. Image courtesy of Harish Mandalika.

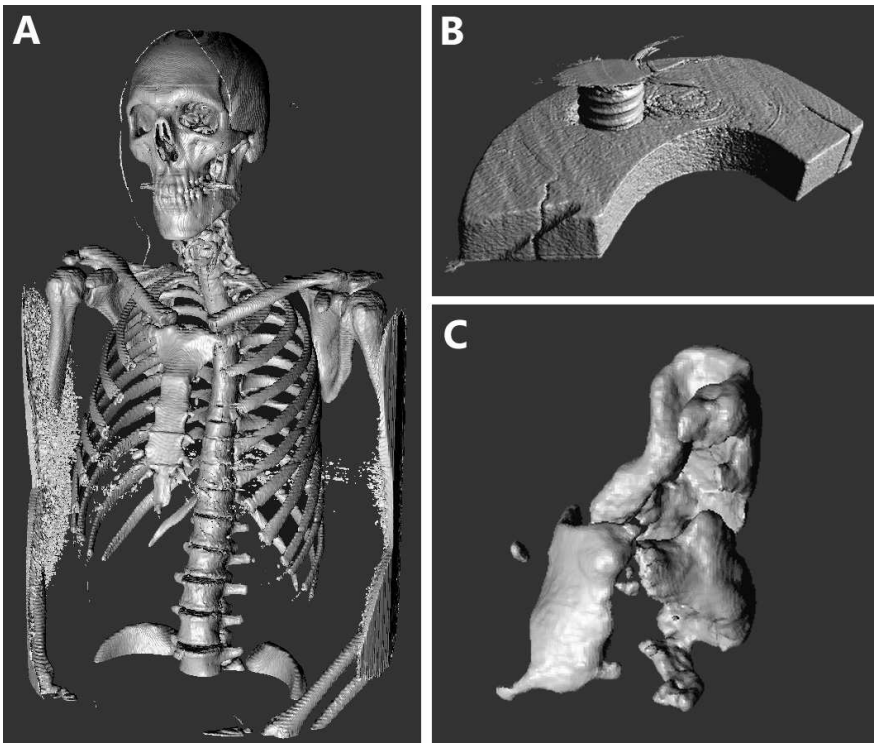


Figure 10.2: Rendering of meshes extracted from three datasets. **A**: Visible Human Male dataset. **B**: one energy volume of the TiScrew dataset. **C**: calcium channel of the Plaque108 dataset.

10.2.3.3 Summary

In conclusion, the speed of rendering and the methods of improving depth perception will need to be examined in greater detail in the future. However, neither is a problem that must be urgently solved, as currently-available techniques are both interactive, and provide sufficient visual quality.

10.2.4 Interaction and tools

This thesis has not focused on developing novel interaction techniques for working with spectral CT datasets. Instead, mouse-based interaction has been extended through the use of the depth buffer (section 5.4), which allows direct interaction with volume data. However, this form of interaction is limited to the visible structures in the dataset; arbitrary placement of points in 3D space is not possible.

This problem stems from using the mouse, a 2D input device, for interacting with a 3D dataset. In the future, it will be beneficial to examine interaction using devices such as a 3D mouse or a stylus, haptic feedback tools such as the Phantom Omni [341], or through hand gesture recognition using devices such as the Microsoft Kinect [342]. This is an area of research that offers numerous possibilities for more natural interaction.

In addition, the work conducted over the course of this research has been aimed at creating a basic set of tools for visualising and analysing currently-available spectral CT datasets generated by the MARS molecular imaging system. Highly-specialised tools (for example, the tools for pre-operative planning, advanced segmentation, and blood vessel tracing or unwrapping) have not been investigated, and remain as topics for future research.

10.2.5 Hybrid 2D and stereoscopic 3D system for spectral CT data visualisation

This system, based on MARS Vision, is currently in development. It uses a stereoscopic 3D display called zSpace [343], which also supports stylus-based interaction and head tracking. The current version is shown in Figure 10.3. It is a multi-monitor setup where 2D slices and most of the GUI for adjusting the visualisation parameters occupy one screen (A), while the meshes extracted from MARS datasets are displayed on the zSpace screen (B). A stylus is used to select and manipulate objects during 3D mesh visualisation (C).

The main aim of this system is the synchronisation of mesh and slice visualisation, using a stylus as an input device. This project is led by Harish Mandalika,

Figure 10.3: Photograph of the hybrid visualisation system. A standard monitor (**A**) is positioned on top, and is used to display 2D slice views and most necessary GUI elements. The zSpace screen (**B**) is used for displaying extracted meshes. The user wears glasses (not shown) that are tracked by the cameras inside zSpace. Interaction is performed using a stylus (**C**). The Mouse12 dataset (section 9.6.1) is shown on the zSpace screen.

a PhD student from the University of Canterbury, while I am currently assisting with the integration of zSpace into MARS Vision and with tool design and implementation.

Prior to visualisation, meshes are extracted from the energy and/or material

volumes of a MARS spectral CT dataset using the Marching Cubes algorithm. Usually, one mesh is extracted per volume, and all meshes are rendered concurrently. The user is able to toggle the visibility of each mesh and examine each one individually.

Mesh colours are based on transfer functions designed by the editor described in Chapter 6 (for example, consider the colours of the meshes in Figure 10.3B, and their similarity to the colours used during DVR in Figure 9.22). In addition, the user is able to move a slicing plane through the dataset using the stylus integrated into the zSpace system. Since both the meshes, and the slice, are movable, the user can choose the best viewing angle and position, and rotate the slice as needed.

The hypothesis is that this interaction style is more natural and easy-to-learn for manipulating objects in 3D space than current mouse-based techniques. Combining it with traditional 2D slice visualisation, as shown in Figure 10.3, may improve the diagnostic accuracy and speed, since the users are able to:

- View the overall structure of the dataset, represented by a set of meshes that show the outlines of all materials. This provides the context.

- Precisely select the slice they wish to view using the stylus. Once the desired slice position and orientation are found, slice data may be viewed on a standard (non-stereo 3D) monitor, as is common in clinical practice.

Therefore, the proposed system introduces novel interaction techniques for interacting with spectral CT data, and combines them with the techniques commonly used for scientific and medical data visualisation.

This work is expected to result in at least two publications. The first, as mentioned above, is focused on comparing the speed and accuracy of diagnosis using the proposed hybrid 2D/3D approach to the traditional 2D-only approach. A user study involving radiologists from Christchurch Hospital is currently ongoing. For the second publication, the hybrid system will be extended to support speech and gesture input, which will again be evaluated through a user study.

10.2.6 Evaluation

This thesis has focused on solving the needs of the current MARS users, and on creating the tools useful for visualising the results of pre-clinical research. These

tools have been well-received and informal feedback has been positive. In addition, several publications have demonstrated their use for visualising the results of preclinical research using the MARS molecular imaging system.

However, formal user studies would help verify their utility and compare them to standard software such as ImageJ, MATLAB, or DICOM viewers. Some tools that may be evaluated in this manner include:

- The simple transfer function editor for material volumes, described in section 6.2, may be compared to traditional graph transfer function editors. This would help establish whether controlling the appearance of channels using a colour scheme based on window and level settings is sufficiently precise and flexible for the purposes of visualising material volumes.

- DVR visualisation with 2D and 3D magic lenses may be compared to standard DVR visualisation. Alternatively, 2D and 3D magic lenses may be compared to each other. Several factors, such as the ease of placement, and the improvement in the visibility of ROIs, may be assessed.

- TIP (section 7.2) can be compared with standard MIP, or other similar intensity projections. In addition, the improvement in depth perception when stereoscopic 3D rendering is used may also be formally evaluated. As noted in section 10.2.3.1, early results suggest that the improvement is noticeable, and that stereoscopic rendering helps restore some depth information that is lost during TIP visualisation.

In addition, other tools may be evaluated using different methods. For example, in the case of the concentration-volume histogram (section 8.1.3), evaluation may involve injecting a small animal (most likely a mouse) with a contrast agent and scanning it multiple times over the course of several days. The amount of information conveyed by the CVHs created after each scan can then be compared to that conveyed by the existing methods (for example, ROI statistics).

10.2.7 Summary

This section has explored the potential areas for further research in the field of spectral CT data visualisation and interaction. The most significant change that can be expected to take place in the near future is the introduction of spectral CT

into clinical practice. This will fundamentally change the composition of the user base, and instigate more research into developing the tools and GUIs suitable for clinical work.

Other areas for future research include faster, higher-quality 3D rendering algorithms, the GUIs and tools for working with tissue type volumes, and novel interaction techniques. The MARS team has already started the development of a hybrid multi-modal interface for spectral CT data visualisation, using MARS Vision, the software I have developed over the course of my PhD research, as a platform.

10.3 Conclusion

This thesis is an in-depth examination of the requirements and tools for effective visualisation of spectral CT datasets. This work has:

- Assessed the current state of the MARS spectral CT data processing toolchain and explained the difference between the energy and material data formats. Each format requires a different approach to visualisation, and a different methodology for designing transfer functions.

- Reviewed the methods used for visualising spectral CT data, as well as the data generated by other medical imaging modalities. The outcome of this review was a set of requirements for visualising current MARS datasets.

- Designed the algorithms for 2D and 3D visualisation of MARS spectral CT data. These algorithms support an arbitrary combination of energy and/or material volumes, and are optimised to run on consumer-level CPU and GPU hardware.

- Created a simplified GUI for creating transfer functions for material volumes, and integrated it with a traditional graph-based transfer function editor. The resulting hybrid transfer function editor can be used for working with both energy, and material volumes of a spectral CT dataset.

- Developed the tools for displaying the regions of interest inside spectral CT datasets along with the context. These tools include 2D and 3D magic lenses, overlays, and the threshold intensity projection.

- Implemented the tools for segmenting, measuring, editing, and annotating MARS datasets. Earlier visualisation tools developed by the MARS team focused primarily on displaying spectral CT data, and did not possess these features.

- Written an application, called MARS Vision, which contains all visualisation tools and techniques developed over the course of this research. This application is fully integrated into the MARS software toolchain, and is currently being used by the pre-clinical researchers from the MARS team.

- Tested MARS Vision by using the spectral CT datasets acquired during the research into the clinical applications of spectral CT. This work has resulted in several publications [4, 9, 11, 13, 155].

10.3.1 Short term outcomes

MARS Vision, which implements the tools and algorithms described in this thesis, is currently being used for visualising the results of studies into contrast agent detection in small animal models, atherosclerotic plaque imaging, and metal artefact reduction. It has also been incorporated into the set of tools supplied with commercially-available MARS systems.

10.3.2 Medium term outcomes

Over the next one to two years, MARS Vision will be extended into a hybrid visualisation system employing novel multi-modal interaction techniques, as described in section 10.2.5. This work is expected to result in several publications for the MARS team. In addition, the multi-volume DVR algorithm presented in this thesis will likely be modified to support level of detail rendering of human-scale MARS datasets.

10.3.3 Long term outcomes

In the long term, the requirements suggested in this thesis may prove useful during the development of clinical spectral CT data visualisation software. This software may be an extension of MARS Vision, or it may be an entirely different application. The data fusion algorithms, tools, and GUI concepts presented in this thesis may

be extended or simplified, depending on the needs of the future clinical users of spectral CT.

10.3.4 Summary

In conclusion, the work presented in this thesis has advanced spectral CT technology by developing novel visualisation algorithms, GUIs for performing data fusion, and tools for displaying ROIs inside spectral CT datasets. These tools have been demonstrated to be suitable for visualising the datasets acquired by the MARS molecular imaging system, and provide a platform for future research and software development.

References

[1] J. Fornaro, S. Leschka, D. Hibbeln, A. Butler, N. Anderson, G. Pache, H. Scheffel, S. Wildemuth, H. Alkadhi, and P. Stolzmann, "Dual- and multi-energy CT: approach to functional imaging," *Insights into Imaging*, vol. 2, pp. 149–159, 2011.

[2] N. G. Anderson and A. P. Butler, "Clinical applications of spectral molecular imaging: potential and challenges," *Contrast Media & Molecular Imaging*, vol. 9, no. 1, pp. 3–12, 2014. [Online]. Available: http://dx.doi.org/10.1002/cmmi.1550

[3] K. Taguchi and J. S. Iwanczyk, "Vision 20/20: Single photon counting x-ray detectors in medical imaging," *Medical Physics*, vol. 40, no. 10, pp. –, 2013. [Online]. Available: http://scitation.aip.org/content/aapm/journal/medphys/40/10/10.1118/1.4820371

[4] R. Aamir, A. Chernoglazov, C. J. Bateman, A. P. H. Butler, P. H. Butler, N. G. Anderson, S. T. Bell, R. K. Panta, J. L. Healy, J. L. Mohr, K. Rajendran, M. F. Walsh, N. de Ruiter, S. P. Gieseg, T. Woodfield, P. F. Renaud, L. Brooke, S. Abdul-Majid, M. Clyne, R. Glendenning, P. J. Bones, M. Billinghurst, C. Bartneck, H. Mandalika, R. Grasset, N. Schleich, N. Scott, S. J. Nik, A. Opie, T. Janmale, D. N. Tang, D. Kim, R. M. Doesburg, R. Zainon, J. P. Ronaldson, N. J. Cook, D. J. Smithies, and K. Hodge, "MARS spectral molecular imaging of lamb tissue: data collection and image analysis," *Journal of Instrumentation*, vol. 9, no. 02, p. P02005, 2014. [Online]. Available: http://stacks.iop.org/1748-0221/9/i=02/a=P02005

[5] D. Cormode, E. Roessl, and A. e. a. Thran, "Atherosclerotic plaque composition: analysis with multicolor CT and targeted gold nanoparticles," *Radiology*, vol. 9, pp. 774–82, 2010.

[6] N. Anderson, A. Butler, N. Scott, N. Cook, J. Butzer, N. Schleich, M. Firsching, R. Grasset, N. de Ruiter, M. Campbell, and P. Butler, "Spectroscopic (multi-energy) ct distinguishes iodine and barium contrast material in mice," *European Radiology*, vol. 20, pp. 2126–2134, 2010, 10.1007/s00330-010-1768-9. [Online]. Available: http://dx.doi.org/10.1007/s00330-010-1768-9

[7] S. Feuerlein, E. Roessl, R. Proksa, G. Martens, O. Klass, M. Jeltsch, V. Rasche, H.-J. Brambs, M. H. K. Hoffmann, and J.-P. Schlomka, "Multienergy

Photon-counting K-edge Imaging: Potential for Improved Luminal Depiction in Vascular Imaging," *Radiology*, vol. 249, no. 3, pp. 1010–1016, 2008, pMID: 18849505. [Online]. Available: http://dx.doi.org/10.1148/radiol.2492080560

[8] P. Lv, X. Lin, K. Chen, and J. Gao, "Spectral CT in patients with small HCC: investigation of image quality and diagnostic accuracy," *European Radiology*, vol. 22, no. 10, pp. 2117–2124, 2012. [Online]. Available: http://dx.doi.org/10.1007/s00330-012-2485-3

[9] K. Rajendran, M. F. Walsh, N. J. A. de Ruiter, A. I. Chernoglazov, R. K. Panta, A. P. H. Butler, P. H. Butler, S. T. Bell, N. G. Anderson, T. B. F. Woodfield, S. J. Tredinnick, J. L. Healy, C. J. Bateman, R. Aamir, R. M. N. Doesburg, P. F. Renaud, S. P. Gieseg, D. J. Smithies, J. L. Mohr, V. B. H. Mandalika, A. M. T. Opie, N. J. Cook, J. P. Ronaldson, S. J. Nik, A. Atharifard, M. Clyne, P. J. Bones, C. Bartneck, R. Grasset, N. Schleich, and M. Billinghurst, "Reducing beam hardening effects and metal artefacts in spectral CT using Medipix3RX," *Journal of Instrumentation*, vol. 9, no. 03, p. P03015, 2014. [Online]. Available: http://stacks.iop.org/1748-0221/9/i=03/a=P03015

[10] P. Baturin, Y. Alivov, and S. Molloi, "Spectral CT imaging of vulnerable plaque with two independent biomarkers," *Physics in Medicine and Biology*, vol. 7, pp. 4117–38, 2012.

[11] R. K. Panta, C. J. Bateman, J. L. Healy, A. Chernoglazov, S. P. Gieseg, A. P. H. Butler, P. H. Butler, and N. G. Anderson, "Implementing Spectral Molecular Imaging (Spectral CT) in Soft Tissue," in *National Conference on Medical Physics AMPICON 2013*, 2013.

[12] D. T. Boll, N. A. Patil, E. K. Paulson, E. M. Merkle, R. C. Nelson, S. T. Schindera, E. Roessl, G. Martens, R. Proksa, T. R. Fleiter, and J.-P. Schlomka, "Focal Cystic High-Attenuation Lesions: Characterization in Renal Phantom by Using Photon-counting Spectral CT – Improved Differentiation of Lesion Composition," *Radiology*, vol. 254, no. 1, pp. 270–276, 2010, pMID: 20032158. [Online]. Available: http://dx.doi.org/10.1148/radiol.09090068

[13] M. F. Walsh, S. J. Nik, S. Procz, M. Pichotka, S. T. Bell, C. J. Bateman, R. M. N. Doesburg, N. de Ruiter, A. I. Chernoglazov, R. K. Panta, A. P. H. Butler, and P. H. Butler, "Spectral CT data acquisition with Medipix3.1," *Journal of Instrumentation*, vol. 8, no. 10, p. P10012, 2013. [Online]. Available: http://stacks.iop.org/1748-0221/8/i=10/a=P10012

[14] S. R. Arridge, "Optical tomography in medical imaging," *Inverse Problems*, vol. 15, no. 2, p. R41, 1999. [Online]. Available: http://stacks.iop.org/0266-5611/15/i=2/a=022

[15] M. Phelps, J. Mazziotta, and H. Schelbert, *Positron emission tomography and autoradiography: Principles and applications for the brain and heart.* Raven Press, New York, USA, 1985.

[16] G. Hounsfield, "Method and Apparatus for Measuring X- or y-Radiation Absorption or Transmission at Plural Angles and Analyzing the Data," United States of America Patent 3 778 614, December, 1973. [Online]. Available: http://www.freepatentsonline.com/3778614.html

[17] E. C. Beckmann, "CT scanning: the early days," *British Journal of Radiology*, vol. 79, pp. 5–8, 2006.

[18] W. A. Kalender, "X-ray computed tomography," *Physics in Medicine and Biology*, vol. 51, no. 13, p. R29, 2006.

[19] S. R. Deans, *The Radon transform and some of its applications.* Courier Dover Publications, 2007.

[20] J. Bushberg, *The essential physics of medical imaging.* Lippincott Williams & Wilkins, 2002. [Online]. Available: http://books.google.com/books?id=jTAwGTYYiusC

[21] F. Natterer, *The Mathematics of Computerized Tomography.* Universitat Munster, 2001.

[22] N. D. Tang, N. de Ruiter, J. L. Mohr, A. P. H. Butler, P. H. Butler, and R. Aamir, "Using Algebraic Reconstruction in Computed Tomography," in *Proceedings of the 27th Conference on Image and Vision Computing New Zealand*, ser. IVCNZ '12. New York, NY, USA: ACM, 2012, pp. 216–221. [Online]. Available: http://doi.acm.org/10.1145/2425836.2425881

[23] J. Hsieh, "Computed tomography: principles, design, artifacts, and recent advances," in *SPIE Bellingham, WA*. SPIE Bellingham, WA, 2009.

[24] A. P. Dhawan, *Medical Image Analysis.* John Wiley & Sons, 2011.

[25] R. E. Alvarez and A. Macovski, "Energy-selective reconstructions in X-ray computerised tomography," *Physics in Medicine and Biology*, vol. 21, no. 5, p. 733, 1976. [Online]. Available: http://stacks.iop.org/0031-9155/21/i=5/a=002

[26] G. D. Chiro, R. A. Brooks, R. M. Kessler, G. S. Johnston, A. E. Jones, J. R. Herdt, and W. T. Sheridan, "Tissue Signatures with Dual-Energy Computed Tomography," *Radiology*, vol. 131, no. 2, pp. 521–523, 1979, pMID: 441344. [Online]. Available: http://dx.doi.org/10.1148/131.2.521

[27] W. A. Kalender, W. H. Perman, J. R. Vetter, and E. Klotz, "Evaluation of a prototype dual-energy computed tomographic apparatus. I. Phantom studies," *Medical Physics*, vol. 13, no. 3, pp. 334–339, 1986. [Online]. Available: http://scitation.aip.org/content/aapm/journal/medphys/13/3/10.1118/1.595958

[28] J. Schoepf and P. Collett, "New Dimensions in Imaging: The Awakening of Dual-Energy CT," *American Journal of Roentgenology*, vol. 199, pp. S1–S2, 2012.

[29] W. Brody, D. Cassel, F. Sommer, L. Lehmann, A. Macovski, R. Alvarez, N. Pelc, S. Riederer, and A. Hall, "Dual-energy projection radiography: initial clinical experience," *American Journal of Roentgenology*, vol. 137, pp. 201–205, 1981.

[30] R. B. Mazess, H. S. Barden, J. P. Bisek, and J. Hanson, "Dual-energy x-ray absorptiometry for total-body and regional bone-mineral and soft-tissue composition." *The American journal of clinical nutrition*, vol. 51, no. 6, pp. 1106–1112, 1990.

[31] H. Kröger, A. Kotaniemi, P. Vainio, and E. Alhava, "Bone densitometry of the spine and femur in children by dual-energy x-ray absorptiometry," *Bone and mineral*, vol. 17, no. 1, pp. 75–85, 1992.

[32] J. Giersch and J. Durst, "Monte Carlo simulations in X-ray imaging," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 591, no. 1, pp. 300 – 305, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168900208004622

[33] E. Roessl and R. Proksa, "K-edge imaging in x-ray computed tomography using multi-bin photon counting detectors," *Physics in Medicine and Biology*, vol. 52, no. 15, p. 4679, 2007. [Online]. Available: http://stacks.iop.org/0031-9155/52/i=15/a=020

[34] M. Andrecut, "Parallel GPU Implementation of Iterative PCA Algorithms," *Journal of Computational Biology*, vol. 16, no. 11, p. 45, 2008. [Online]. Available: http://arxiv.org/abs/0811.1081

[35] P. Sukovle and N. Clinthorne, "Basis material decomposition using triple-energy x-ray computed tomography," in *Instrumentation and Measurement Technology*

Conference, 1999. IMTC/99. Proceedings of the 16th IEEE, vol. 3, 1999, pp. 1615–1618 vol.3.

[36] C. J. Bateman, "Methods for Material Discrimination in MARS Multi-Energy CT," Ph.D. dissertation, University of Otago, 2015.

[37] R. Ballabriga, M. Campbell, E. H. M. Heijne, X. Llopart, and L. Tlustos, "The Medipix3 Prototype, a Pixel Readout Chip Working in Single Photon Counting Mode With Improved Spectrometric Performance," *IEEE Transactions on Nuclear Science*, vol. 54, pp. 1824–1829, Oct. 2007.

[38] L. G. Brown, "A survey of image registration techniques," *ACM Comput. Surv.*, vol. 24, no. 4, pp. 325–376, Dec. 1992. [Online]. Available: http://doi.acm.org/10.1145/146370.146374

[39] T. Lange, N. Papenberg, S. Heldmann, J. Modersitzki, B. Fischer, H. Lamecker, and P. Schlag, "3D ultrasound-CT registration of the liver using combined landmark-intensity information," *International Journal of Computer Assisted Radiology and Surgery*, vol. 4, no. 1, pp. 79–88, 2009. [Online]. Available: http://dx.doi.org/10.1007/s11548-008-0270-1

[40] R. Hamilton, M. Blend, C. Pelizzari, B. Milliken, and S. Vijayakumar, "Using vascular structure for CT-SPECT registration in the pelvis," *Journal of nuclear medicine : official publication, Society of Nuclear Medicine*, vol. 40, pp. 347–351, 1999.

[41] J.-M. Rouet, J.-J. Jacq, and C. Roux, "Genetic algorithms for a robust 3-D MR-CT registration," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 4, no. 2, pp. 126–136, June 2000.

[42] S. Nicolaou, T. Liang, D. T. Murphy, J. R. Korzan, H. Ouellette, and P. Munk, "Dual-Energy CT: A Promising New Technique for Assessment of the Musculoskeletal System," *Am. J. Roentgenol.*, vol. 199, pp. S78–S86, 2012.

[43] H. K. Choi, L. C. Burns, K. Shojania, N. Koenig, G. Reid, M. Abufayyah, G. Law, A. S. Kydd, H. Ouellette, and S. Nicolaou, "Dual energy CT in gout: a prospective validation study," *Annals of the rheumatic diseases*, 2012.

[44] G. Ascenti, S. Mazziotti, S. Lamberto, A. Bottari, S. Caloggero, S. Racchiusa, A. Mileto, and E. Scribano, "Dual-energy CT for detection of endoleaks after endovascular abdominal aneurysm repair: usefulness of colored iodine overlay." *American Journal of Roentgenology*, vol. 196, pp. 1408–14, 2011.

[45] C. N. De Cecco, A. Darnell, M. Rengo, G. Muscogiuri, D. Bellini, C. Ayuso, and A. Laghi, "Dual-Energy CT: Oncologic Applications," *American Journal of Roentgenology*, vol. 199, pp. S98–S105, 2012.

[46] R. Johnson, Thorsten, B. Krauss, M. Sedlmair, M. Grasruck, H. Bruder, D. Morhard, C. Fink, S. Weckbach, M. Lenhard, B. Schmidt, T. Flohr, F. Reiser, Maximilian, and C. Becker, "Material differentiation by dual energy CT: initial experience," *European Radiology*, vol. 17, no. 6, pp. 1510–1517, 2007.

[47] G. Hidas, R. Eliahou, M. Duvdevani, P. Coulon, L. Lemaitre, O. N. Gofrit, D. Pode, and J. Sosna, "Determination of Renal Stone Composition with Dual-Energy CT: In Vivo Analysis and Comparison with X-ray Diffraction 1," *Radiology*, vol. 257, no. 2, pp. 394–401, 2010.

[48] J. P. Schlomka, E. Roessl, R. Dorscheid, S. Dill, G. Martens, T. Istel, C. Baeumer, C. Herrmann, R. Steadman, G. Zeitler, A. Livne, and R. Proksa, "Experimental feasibility of multi-energy photon-counting K-edge imaging in pre-clinical computed tomography," *Physics in Medicine and Biology*, vol. 53, no. 15, p. 4031, 2008. [Online]. Available: http://stacks.iop.org/0031-9155/53/i=15/a=002

[49] J. Butzer, A. Butler, N. Cook, P. Butler, F. Ross, N. Schleich, J. Selkirk, R. Watts, J. Meyer, N. Scott, P. Bones, D. van Leeuwen, S. Hemmingsen, T. Melzer, N. Anderson, and MARS-CT Team, "MARS: a 3D Spectroscopic X-Ray Imaging Device Based on Medipix," in *IEEE Nuclear Science Symposium and Medical Imaging Conference, 19-25 Oct 2008.* University of Canterbury. Electrical and Computer Engineering, 2008.

[50] S. Kappler, A. Henning, B. Krauss, F. Schoeck, K. Stierstorfer, T. Weidinger, and T. Flohr, "Multi-energy performance of a research prototype CT scanner with small-pixel counting detector," in *Proc. SPIE 8668, Medical Imaging 2013: Physics of Medical Imaging, 86680O*, 2013.

[51] J. Iwanczyk, E. Nygard, O. Meirav, J. Arenson, W. C. Barber, N. Hartsough, N. Malakhov, and J. Wessel, "Photon Counting Energy Dispersive Detector Arrays for X-ray Imaging," *Nuclear Science, IEEE Transactions on*, vol. 56, no. 3, pp. 535–542, June 2009.

[52] X. Wang, D. Meier, K. Taguchi, D. J. Wagenaar, B. E. Patt, and E. C. Frey, "Material separation in x-ray CT with energy resolved photon-counting detectors," *Medical Physics*, vol. 38, no. 3, pp. 1534–1546, 2011. [Online]. Available: http://scitation.aip.org/content/aapm/journal/medphys/38/3/10.1118/1.3553401

[53] A. Butler, N. Cook, N. Schleich, J. Butzer, P. Bones, P. Butler, and MARS-CT Team, "Processing of Spectral X-ray Data Using Principal Components Analysis," *Nuclear Instruments and Methods in Physics Research Section A*, vol. 633, pp. S140–S142, 2010.

[54] J. S. Butzer, "MARS-CT: Biomedical Spectral X-ray Imaging with Medipix," MSc, University of Canterbury. Physics and Astronomy, 2009.

[55] R. Zainon, J. Ronaldson, T. Janmale, N. Scott, T. Buckenham, A. Butler, P. Butler, R. Doesburg, S. Gieseg, J. Roake, and N. Anderson, "Spectral CT of carotid atherosclerotic plaque: comparison with histology," *European Radiology*, vol. 22, no. 12, pp. 2581–2588, 2012. [Online]. Available: http://dx.doi.org/10.1007/s00330-012-2538-7

[56] P. Ghosh, G. Han, M. De, C. K. Kim, and V. M. Rotello, "Gold nanoparticles in delivery applications," *Advanced drug delivery reviews*, vol. 60, no. 11, pp. 1307–1315, 2008.

[57] R. Zainon, A. P. H. Butler, N. J. Cook, J. S. Butzer, N. Schleich, N. D. Ruiter, L. Tlustos, M. J. Clark, R.Heinz, and P. H. Butler, "Construction and Operation of the MARS-CT Scanner," *Internetworking Indonesia Journal*, vol. 2, pp. 2–10, 2010.

[58] M. F. Walsh, A. M. T. Opie, J. P. Ronaldson, R. M. N. Doesburg, S. J. Nik, J. L. Mohr, R. Ballabriga, A. P. H. Butler, and P. H. Butler, "First CT using Medipix3 and the MARS-CT-3 spectral scanner," *Journal of Instrumentation*, vol. 6, no. 01, p. C01095, 2011. [Online]. Available: http://stacks.iop.org/1748-0221/6/i=01/a=C01095

[59] N. de Ruiter, "Algorithms to process data from the MARS molecular imaging system," Ph.D. dissertation, University of Otago, 2015.

[60] M. Hadwiger, J. M. Kniss, C. Rezk-salama, D. Weiskopf, and K. Engel, *Real-time Volume Graphics*. Natick, MA, USA: A. K. Peters, Ltd., 2006.

[61] P. Lacroute and M. Levoy, "Fast volume rendering using a shear-warp factorization of the viewing transformation," in *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH '94. New York, NY, USA: ACM, 1994, pp. 451–458. [Online]. Available: http://doi.acm.org/10.1145/192161.192283

[62] D. Laur and P. Hanrahan, "Hierarchical splatting: a progressive refinement algorithm for volume rendering." in *SIGGRAPH'91*, 1991, pp. 285–288.

[63] Merge Healthcare. (2011) http://www.merge.com. [Online]. Available: http://www.merge.com

[64] K. Fridell, L. Edgren, L. Lindskold, P. Aspelin, and N. Lundberg, "The Impact of PACS on Radiologists' Work Practice," *Journal of Digital Imaging*, vol. 20, pp. 411–421, 2007.

[65] National Electrical Manufacturers Association and American College of Radiology and others, *Digital imaging and communications in medicine (DICOM)*. National Electrical Manufacturers Association, 1998.

[66] Intelerad Medical Systems Incorporated. (2015) Intelerad Medical Systems Incorporated. http://www.intelerad.com/. [Online]. Available: http://www.intelerad.com/

[67] National Institute of Health. (2014) Imagej. http://imagej.nih.gov/ij/.

[68] A. Fedorov, R. Beichel, and K.-C. J. et al., "3D Slicer as an Image Computing Platform for the Quantitative Imaging Network," *Magn Reson Imaging*, vol. 30, pp. 1323–41, 2012.

[69] I. Wolf, M. Vetter, I. Wegner, M. Nolden, T. Böttger, M. Hastenteufel, M. Schöbinger, T. Kunert, H. Meinzer, and Deutsches Krebsforschungszentrum, "The Medical Imaging Interaction Toolkit (MITK) a toolkit facilitating the creation of interactive software by extending VTK and ITK," www.mitk.org.

[70] R. Chityala, S. Pudipeddi, L. Arensten, and S. Hui, "Segmentation and Visualization of a Large, High-Resolution Micro-CT Data of Mice," *Journal of Digital Imaging*, vol. 26, pp. 302–308, 2013.

[71] NVIDIA Corporation. (2014, July) Product Spects - Quadro GPUs for Desktop PCs. NVIDIA Corporation. [Online]. Available: http://www.nvidia.com/object/quadro-desktop-gpus-specs.html

[72] ——. (2014, July) Geforce gtx 780 ti — specifications. NVIDIA Corporation. [Online]. Available: http://www.geforce.com/hardware/desktop-gpus/geforce-gtx-780-ti/specifications

[73] Advanced Micro Devices, Inc. (2014, July) Amd radeon r9 series graphics. http://www.amd.com/en-us/products/graphics/desktop/r9. Advanced Micro Devices, Inc. [Online]. Available: http://www.amd.com/en-us/products/graphics/desktop/r9

[74] A. Chernoglazov, "Improving Visualisation of Large Multi-Variate Datasets: New Hardware-Based Algorithms and Rendering Techniques," Master's thesis, University of Canterbury, 2012.

[75] J. Sindhu, A. Poh, T. Lim, E. Chan, and L. Chong, "The iPad Tablet Computer for Mobile On-Call Radiology Diagnosis? Auditing Discrepancy in CT and MRI Reporting," *Journal of Digital Imaging*, vol. 25, pp. 628–634, 2012.

[76] P. McLaughlin, S. O. Neill, N. Fanning, A. M. McGarrigle, O. J. Connor, G. Wuse, and M. M. Maher, "Emergency CT brain: preliminary interpretation with a tablet device: image quality and diagnostic performance of the Apple iPad." *Emergency Radiology*, vol. 19, pp. 127–33, 2012.

[77] Friedrich-Alexander Universitaet Erlangen-Nuernberg. (2015) The Volume Library. [Online]. Available: http://lgdv.cs.fau.de/External/vollib/

[78] M. A. van Ooijen, Peter, P. Koesoema, Allya, and M. Oudkerk, "User Questionnaire to Evaluate the Radiological Workspace," *Journal of Digital Imaging*, vol. 19, no. 1, pp. 52–59, 2006. [Online]. Available: http://dx.doi.org/10.1007/s10278-006-0629-1

[79] T. Boehm, O. Handgraetinger, J. Link, R. Ploner, D. Voellmy, B. Marincek, and S. Wildermuth, "Evaluation of radiological workstations and web-browser-based image distribution clients for a PACS project in hands-on workshops," *European Radiology*, vol. 14, no. 5, pp. 908–914, 2004. [Online]. Available: http://dx.doi.org/10.1007/s00330-003-2205-0

[80] S. Langer and J. Wang, "An evaluation of ten digital image review workstations," *Journal of Digital Imaging*, vol. 10, no. 2, pp. 65–78, 1997. [Online]. Available: http://dx.doi.org/10.1007/BF03168558

[81] R. Honea, C. McCluggage, B. Parker, D. ONeall, and K. Shook, "Evaluation of commercial PC-based DICOM image viewer," *Journal of Digital Imaging*, vol. 11, no. 1, pp. 151–155, 1998. [Online]. Available: http://dx.doi.org/10.1007/BF03168289

[82] F. Liu, P. Liang, X. Yu, T. Lu, Z. Cheng, C. Lei, and Z. Han, "A three-dimensional visualisation preoperative treatment planning system in microwave ablation for liver cancer: A preliminary clinical application," *International Journal of Hyperthermia*, vol. 29, no. 7, pp. 671–677, 2013.

[83] R. Gerrah, D. M. Bardo, R. D. Reed, R. E. Sunstrom, and S. M. Langley, "Adjustment of the surgical plan in repair of congenital heart disease: The power of cross-sectional imaging and three-dimensional visualization," *Congenital heart disease*, vol. 9, no. 1, pp. E31–E36, 2014.

[84] W. You, D. Wang, G. Liu, X.-c. LI, J.-y. XIONG, X.-l. WU, and M. LI, "The application of digital orthopedic 3d visualization technology in preoperative planning of the proximal humeral comminuted fracture," *Chin J Mod Oper Surg*, vol. 17, pp. 25–28, 2013.

[85] M. J. Zinser, R. A. Mischkowski, T. Dreiseidler, O. C. Thamm, D. Rothamel, and J. E. Zöller, "Computer-assisted orthognathic surgery: waferless maxillary positioning, versatility, and accuracy of an image-guided visualisation display," *British Journal of Oral and Maxillofacial Surgery*, vol. 51, no. 8, pp. 827–833, 2013.

[86] L. Maier-Hein, P. Mountney, A. Bartoli, H. Elhawary, D. Elson, A. Groch, A. Kolb, M. Rodrigues, J. Sorger, S. Speidel, and D. Stoyanov, "Optical techniques for 3d surface reconstruction in computer-assisted laparoscopic surgery," *Medical Image Analysis*, vol. 17, no. 8, pp. 974 – 996, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1361841513000480

[87] J. Totz, K. Fujii, P. Mountney, and G.-Z. Yang, "Enhanced visualisation for minimally invasive surgery," *International journal of computer assisted radiology and surgery*, vol. 7, no. 3, pp. 423–432, 2012.

[88] R. A. Drebin, L. Carpenter, and P. Hanrahan, "Volume Rendering," *SIGGRAPH Comput. Graph.*, vol. 22, no. 4, pp. 65–74, Jun. 1988. [Online]. Available: http://doi.acm.org/10.1145/378456.378484

[89] S. D. Roth, "Ray casting for modeling solids," *Computer graphics and image processing*, vol. 18, no. 2, pp. 109–144, 1982.

[90] C. D. Hansen and C. R. Johnson, *Visualization handbook*. Academic Press, 2011.

[91] D. Jönsson, E. Sundén, A. Ynnerman, and T. Ropinski, "A Survey of Volumetric Illumination Techniques for Interactive Volume Rendering," *Computer*

Graphics Forum, vol. 33, no. 1, pp. 27–51, 2014. [Online]. Available: http://dx.doi.org/10.1111/cgf.12252

[92] L. Westover, "Footprint evaluation for volume rendering," *SIGGRAPH Comput. Graph.*, vol. 24, pp. 367–376, September 1990. [Online]. Available: http://doi.acm.org/10.1145/97880.97919

[93] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," *COMPUTER GRAPHICS*, vol. 21, no. 4, pp. 163–169, 1987.

[94] M. Levoy, "Display of surfaces from volume data," *Computer Graphics and Applications, IEEE*, vol. 8, no. 3, pp. 29 –37, may 1988.

[95] D. Ruijters, B. M. ter Haar-Romeny, and P. Suetens, "Accuracy of GPU-based B-spline evaluation," in *Proceedings of the Tenth IASTED International Conference on Computer Graphics and Imaging*, ser. CGIM '08. Anaheim, CA, USA: ACTA Press, 2008, pp. 117–122. [Online]. Available: http://dl.acm.org/citation.cfm?id=1722302.1722327

[96] T. Möller, K. Mueller, Y. Kurzion, R. Machiraju, and R. Yagel, "Design of accurate and smooth filters for function and derivative reconstruction," in *Proceedings of the 1998 IEEE symposium on Volume visualization*, ser. VVS '98. New York, NY, USA: ACM, 1998, pp. 143–151. [Online]. Available: http://doi.acm.org/10.1145/288126.288189

[97] S. Marschner and R. Lobb, "An evaluation of reconstruction filters for volume rendering," in *Visualization, 1994., Visualization '94, Proceedings., IEEE Conference on*, oct 1994, pp. 100 –107, CP10.

[98] R. Fernando, *GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics*. Addison-Wesley, 2004, ch. Chapter 39. Volume Rendering Techniques.

[99] H. Pfister, B. Lorensen, C. Bajaj, G. Kindlmann, W. Schroeder, L. S. Avila, K. Raghu, R. Machiraju, and J. Lee, "The transfer function bake-off," *Computer Graphics and Applications, IEEE*, vol. 21, no. 3, pp. 16–22, 2001.

[100] C. Rezk-Salama and A. Kolb, "Opacity Peeling for Direct Volume Rendering," *Computer Graphics Forum*, vol. 25, no. 3, pp. 597–606, 2006. [Online]. Available: http://dx.doi.org/10.1111/j.1467-8659.2006.00979.x

[101] J. F. Blinn, "Models of light reflection for computer synthesized pictures," *SIGGRAPH Comput. Graph.*, vol. 11, pp. 192–198, July 1977. [Online]. Available: http://doi.acm.org/10.1145/965141.563893

[102] L. Zheng, A. Chaudhari, R. Badawi, and K.-L. Ma, "Using Global Illumination in Volume Visualization of Rheumatoid Arthritis CT Data," *Computer Graphics and Applications, IEEE*, vol. 34, no. 6, pp. 16–23, Nov 2014.

[103] T. Kroes, F. Post, and C. Botha, "Exposure Render: An Interactive Photo-Realistic Volume Rendering Framework," *PLoS ONE*, vol. 7, 2012.

[104] J. Kruger and R. Westermann, "Acceleration techniques for GPU-based volume rendering," in *Proc. IEEE Visualization VIS 2003*, 2003, pp. 287–292.

[105] R. Yagel and Z. Shi, "Accelerating volume animation by space-leaping," in *Visualization, 1993. Visualization '93, Proceedings., IEEE Conference on*, oct 1993, pp. 62 –69.

[106] B. Kainz, M. Grabner, A. Bornik, S. Hauswiesner, J. Muehl, and D. Schmalstieg, "Ray casting of multiple volumetric datasets with polyhedral boundaries on manycore GPUs," in *ACM SIGGRAPH Asia 2009 papers*, ser. SIGGRAPH Asia '09. New York, NY, USA: ACM, 2009, pp. 152:1–152:9. [Online]. Available: http://doi.acm.org/10.1145/1661412.1618498

[107] J. Meyer-Spradow, T. Ropinski, J. Mensmann, and K. Hinrichs, "Voreen: A Rapid-prototyping Environment for Ray-casting-based Volume Visualizations," *IEEE Comput. Graph. Appl.*, vol. 29, no. 6, pp. 6–13, Nov. 2009. [Online]. Available: http://dx.doi.org/10.1109/MCG.2009.130

[108] J. Mensmann, T. Ropinski, and K. H. Hinrichs, "An advanced volume raycasting technique using gpu stream processing," in *GRAPP: International Conference on Computer Graphics Theory and Applications*. Angers: INSTICC Press, 2010, pp. 190–198. [Online]. Available: http://viscg.uni-muenster.de/publications/2010/MRH10

[109] Meshlab. (2014) Meshlab. meshlab.sourceforge.net.

[110] Blender - Free and Open 3D Creation Software. (2014) www.blender.org.

[111] F. Rengier, A. Mehndiratta, H. von Tengg-Kobligk, C. Zechmann, R. Unterhinninghofen, H.-U. Kauczor, and F. Giesel, "3D printing based on imaging data: review of medical applications," *International Journal of Computer Assisted*

Radiology and Surgery, vol. 5, no. 4, pp. 335–341, 2010. [Online]. Available: http://dx.doi.org/10.1007/s11548-010-0476-x

[112] J. Chhugani, S. Vishwanath, J. Cohen, and S. Kumar, "Isoslider: a system for interactive exploration of isosurfaces," in *VisSym*, 2003.

[113] G. ElMasry, D.-W. Sun, and P. Allen, "Near-infrared hyperspectral imaging for predicting colour, ph and tenderness of fresh beef," *Journal of Food Engineering*, vol. 110, no. 1, pp. 127–140, 2012.

[114] A. Gowen, C. O'Donnell, P. Cullen, G. Downey, and J. Frias, "Hyperspectral imaging–an emerging process analytical tool for food quality and safety control," *Trends in Food Science & Technology*, vol. 18, no. 12, pp. 590–598, 2007.

[115] Y. Roggo, A. Edmond, P. Chalus, and M. Ulmschneider, "Infrared hyperspectral imaging for qualitative analysis of pharmaceutical solid forms," *Analytica Chimica Acta*, vol. 535, no. 1, pp. 79–87, 2005.

[116] J. Ellis, H. Davis, and J. Zamudio, "Exploring for onshore oil seeps with hyperspectral imaging," *Oil and Gas Journal*, vol. 99, no. 37, pp. 49–58, 2001.

[117] E. Cloutis, "Review article hyperspectral geological remote sensing: evaluation of analytical techniques," *International Journal of Remote Sensing*, vol. 17, no. 12, pp. 2215–2242, 1996.

[118] D. Stein, J. Schoonmaker, and E. Coolbaugh, "Hyperspectral imaging for intelligence, surveillance, and reconnaissance," DTIC Document, Tech. Rep., 2001.

[119] P. W. Yuen and M. Richardson, "An introduction to hyperspectral imaging and its application for security, surveillance and target acquisition," *The Imaging Science Journal*, vol. 58, no. 5, pp. 241–253, 2010.

[120] R. Fuchs and H. Hauser, "Visualization of multi-variate scientific data," in *Computer Graphics Forum*, vol. 28, no. 6. Wiley Online Library, 2009, pp. 1670–1690.

[121] J. Blaas, C. P. Botha, and F. H. Post, "Interactive visualization of multi-field medical data using linked physical and feature-space views." in *EuroVis*, 2007, pp. 123–130.

[122] A. Amirkhanov, J. Kastner, C. Heinzl, E. Gröller, and B. Fröhler, "Multimodal visualization and analysis of spectral and scalar data," 2015.

[123] I. Viola, A. Kanitsar, and E. Gröller, "Importance-driven volume rendering," in *Proceedings of IEEE Visualization'04*, 2004, pp. 139–145.

[124] S. J. Kim, S. Zhuo, F. Deng, C.-W. Fu, and M. Brown, "Interactive visualization of hyperspectral images of historical documents," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 16, no. 6, pp. 1441–1448, Nov 2010.

[125] P. Rheingans and D. Ebert, "Volume illustration: Nonphotorealistic rendering of volume models," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 7, no. 3, pp. 253–264, 2001.

[126] J. C. Anderson, L. J. Gosink, M. A. Duchaineau, and K. I. Joy, "Interactive visualization of function fields by range-space segmentation," in *Computer Graphics Forum*, vol. 28, no. 3. Wiley Online Library, 2009, pp. 727–734.

[127] L. Nyland, M. Harris, and J. Prins, *GPU Gems 3*. Addison-Wesley, 2007, ch. Fast N-Body Simulation with CUDA, pp. 677–695.

[128] L. Marsalek, A. Hauber, and P. Slusallek, "High-speed volume ray casting with CUDA," in *IEEE Symposium on Interactive Ray Tracing, 2008. RT 2008.*, aug. 2008, p. 185.

[129] N. de Ruiter, "GPU Accelerated Intermixing as a Framework for Interactively Visualizing Spectral CT Data," Master's thesis, University of Canterbury, 2011.

[130] Y.-L. Huang, Y.-C. Shen, and J.-L. Wu, "Scalable computation for spatially scalable video coding using NVIDIA CUDA and multi-core CPU," in *Proceedings of the 17th ACM international conference on Multimedia*, ser. MM '09. New York, NY, USA: ACM, 2009, pp. 361–370. [Online]. Available: http://doi.acm.org/10.1145/1631272.1631323

[131] N. K. Govindaraju, B. Lloyd, Y. Dotsenko, B. Smith, and J. Manferdelli, "High performance discrete Fourier transforms on graphics processors," in *Proceedings of the 2008 ACM/IEEE conference on Supercomputing*, ser. SC '08. Piscataway, NJ, USA: IEEE Press, 2008, pp. 2:1–2:12. [Online]. Available: http://dl.acm.org/citation.cfm?id=1413370.1413373

[132] M. Silberstein, A. Schuster, D. Geiger, A. Patney, and J. D. Owens, "Efficient computation of sum-products on GPUs through software-managed cache," in *Proceedings of the 22nd annual international conference on Supercomputing*, ser. ICS '08. New York, NY, USA: ACM, 2008, pp. 309–318. [Online]. Available: http://doi.acm.org/10.1145/1375527.1375572

[133] B. Jang, D. Kaeli, S. Do, and H. Pien, "Multi GPU implementation of iterative tomographic reconstruction algorithms," in *Biomedical Imaging: From Nano to*

Macro, 2009. ISBI '09. IEEE International Symposium on, 28 2009-july 1 2009, pp. 185 –188.

[134] NVIDIA Corporation. (2011, March) Cuda programming guide (version 4.0). http://developer.download.nvidia.com/compute/cuda/4_0_rc2/toolkit/docs/CUDA_C_Programming_Guide.pdf. NVIDIA Corporation. [Online]. Available: http://developer.download.nvidia.com/compute/cuda/4_0_rc2/toolkit/docs/CUDA_C_Programming_Guide.pdf

[135] Benchmark Reviews. (2010, January) Nvidia fermi architecture. http://benchmarkreviews.com/index.php?option=com_content&task=view&id=440&Itemid=63&limit=1&limitstart=3.

[136] NVIDIA Corporation. (2008) Nvidia's next generation compute architecture: Fermi. http://www.nvidia.com/content/PDF/fermi_white_papers/NVIDIAFermiComputeArchitectureWhitepaper.pdf. NVIDIA Corporation. [Online]. Available: http://www.nvidia.com/content/PDF/fermi_white_papers/NVIDIAFermiComputeArchitectureWhitepaper.pdf

[137] J. D. Owens, D. Luebke, N. Govindaraju, M. Harris, J. Krger, A. Lefohn, and T. J. Purcell, "A Survey of General-Purpose Computation on Graphics Hardware," *Computer Graphics Forum*, vol. 26, no. 1, pp. 80–113, 2007. [Online]. Available: http://www.blackwell-synergy.com/doi/pdf/10.1111/j.1467-8659.2007.01012.x

[138] M. Segal and K. Akeley, *The OpenGL Graphics System: A Specication (Version 3.2 (Core Prole))*, 3rd ed., Khronos Group, December 2009.

[139] K. Gray, *Microsoft DirectX 9 programmable graphics pipeline*. Microsoft Press, 2003.

[140] M. Weiler, M. Kraus, M. Merz, and T. Ertl, "Hardware-based ray casting for tetrahedral meshes," in *Proc. IEEE Visualization VIS 2003*, 2003, pp. 333–340.

[141] L. Weiguo, B. Schmidt, G. Voss, and W. Muller-Wittig, "Streaming Algorithms for Biological Sequence Alignment on GPUs," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 18, no. 9, pp. 1270 –1281, sept. 2007.

[142] K. Perumalla, "Discrete-event Execution Alternatives on General Purpose Graphical Processing Units (GPGPUs)," in *Principles of Advanced and Distributed Simulation, 2006. PADS 2006. 20th Workshop on*, 2006, pp. 74 –81.

[143] Khronos Group. (2011) OpenCL - The open standard for parallel programming of heteregeneous systems. http://www.khronos.org/opencl/.

[144] The Khronos Group. (2015) Opengl shading language. https://www.opengl.org/documentation/glsl/.

[145] R. J. Rost, B. M. Licea-Kane, D. Ginsburg, J. M. Kessenich, B. Lichtenbelt, H. Malan, and M. Weiblen, *OpenGL shading language*. Pearson Education, 2009.

[146] Microsoft Corporation. (2015) Programming guide for hlsl (windows). https://msdn.microsoft.com/en-us/library/windows/desktop/bb509635(v=vs.85).aspx.

[147] P. Hanrahan and J. Lawson, "A Language for Shading and Lighting Calculations," *SIGGRAPH Comput. Graph.*, vol. 24, no. 4, pp. 289–298, Sep. 1990. [Online]. Available: http://doi.acm.org/10.1145/97880.97911

[148] NVIDIA Corporation. (2015) Cuda toolkit 2.0 (august 2008). https://developer.nvidia.com/cuda-toolkit-20-august-2008.

[149] ——. (2011) Cuda toolkit 4.0. http://developer.nvidia.com/cuda-toolkit-40. NVIDIA Corporation. [Online]. Available: http://developer.nvidia.com/cuda-toolkit-40

[150] M. Smelyanskiy, D. Holmes, J. Chhugani, A. Larson, D. M. Carmean, D. Hanson, P. Dubey, K. Augustine, D. Kim, A. Kyker, V. W. Lee, A. D. Nguyen, L. Seiler, and R. Robb, "Mapping High-Fidelity Volume Rendering for Medical Imaging to CPU, GPU and Many-Core Architectures," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1563–1570, Nov. 2009. [Online]. Available: http://dx.doi.org/10.1109/TVCG.2009.164

[151] IMAGE Information Systems. (2015) iQ-VIEW. [Online]. Available: http://www.image-systems.biz/iq-view

[152] Octopus Software. (2015) Octopus Imaging. [Online]. Available: http://www.octopusimaging.eu/en/home

[153] *MATLAB*. Natick, Massachusetts: The MathWorks Inc., Natick, MA, 2010.

[154] Adobe Systems Incorporated. (2015) Developer resources. [Online]. Available: http://partners.adobe.com/public/developer/tiff/index.html

[155] K. Rajendran, S. Tredinnick, N. de Ruiter, A. Chernoglazov, T. Woodfield, A. Butler, and N. Anderson, "Metal Artefact Reduction in Ortohopaedic Implants Using MARS Spectral CT." in *To be presented during the 21st Annual Scientific Meeting of the Australian & New Zealand Orthopaedic Research Society (ANZORS)*, 2015.

[156] DICOM Standards Committee, Working Group 6. (2011) Digital Imaging and Communications in Medicine (DICOM) Supplement 96: Unified Worklist and Procedure Step. [Online]. Available: ftp://medical.nema.org/medical/dicom/final/sup96_ft.pdf

[157] S. Leng, L. Yu, J. Wang, J. G. Fletcher, C. A. Mistretta, and C. H. McCollough, "Noise reduction in spectral CT: Reducing dose and breaking the trade-off between image noise and energy bin selection," *Medical Physics*, vol. 38, no. 9, pp. 4946–4957, 2011. [Online]. Available: http://scitation.aip.org/content/aapm/journal/medphys/38/9/10.1118/1.3609097

[158] M. A. Hurrell, A. P. H. Butler, N. J. Cook, P. H. Butler, J. P. Ronaldson, and R. Zainon, "Spectral Hounsfield units: a new radiological concept," *European Radiology*, vol. 22, pp. 1008–1013, 2012.

[159] F. E. Boas and D. Fleischmann, "CT artifacts: causes and reduction techniques," *Imaging in Medicine*, vol. 4, pp. 229–240, 2012.

[160] W. R. Hendee and F. M. Edwards, "{ALARA} and an integrated approach to radiation protection," *Seminars in Nuclear Medicine*, vol. 16, no. 2, pp. 142 – 150, 1986, nuclear Medicine and the EnvironmentPart I. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0001299886800277

[161] P. M. Joseph and R. D. Spital, "The effects of scatter in x-ray computed tomography," *Medical physics*, vol. 9, no. 4, pp. 464–472, 1982.

[162] V. Spitzer, A. M. J., A. L. Scherzinger, and D. Whitlock, "The Visible Human Male: a technical report," *J Am Med Inform Assoc*, vol. 3, pp. 118–130, 1996.

[163] R. M. N. Doesburg, T. Koenig, S. J. Nik, S. T. Bell, J. P. Ronaldson, M. F. Walsh, A. P. H. Butler, and P. H. Butler, "Spectrum measurement using Medipix3 in Charge Summing Mode," *Journal of Instrumentation*, vol. 7, no. 11, p. C11004, 2012. [Online]. Available: http://stacks.iop.org/1748-0221/7/i=11/a=C11004

[164] L. J. Higgins and M. G. Pomper, "The evolution of imaging in cancer: current state and future challenges," in *Seminars in oncology*, vol. 38, no. 1. Elsevier, 2011, pp. 3–15.

[165] D. W. Townsend, "Combined Positron Emission TomographyComputed Tomography: The Historical Perspective," *Seminars in Ultrasound, {CT} and {MRI}*, vol. 29, no. 4, pp. 232 – 235, 2008, pET/CT. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0887217108000449

[166] P. S. Conti, D. L. Lilien, K. Hawley, J. Keppler, S. T. Grafton, and J. R. Bading, "PET and [18 F]-FDG in oncology: a clinical update," *Nuclear medicine and biology*, vol. 23, no. 6, pp. 717–735, 1996.

[167] A. K. Buck, K. Herrmann, T. Stargardt, T. Dechow, B. J. Krause, and J. Schreyögg, "Economic evaluation of PET and PET/CT in oncology: evidence and methodologic approaches," *Journal of nuclear medicine technology*, vol. 38, no. 1, pp. 6–17, 2010.

[168] M. H. Umbehr, M. Müntener, T. Hany, T. Sulser, and L. M. Bachmann, "The role of 11C-choline and 18F-fluorocholine positron emission tomography (PET) and PET/CT in prostate cancer: a systematic review and meta-analysis," *European urology*, vol. 64, no. 1, pp. 106–117, 2013.

[169] C.-Y. Lin, J.-H. Chen, J.-A. Liang, C.-C. Lin, L.-B. Jeng, and C.-H. Kao, "18f-fdg pet or pet/ct for detecting extrahepatic metastases or recurrent hepatocellular carcinoma: a systematic review and meta-analysis," *European journal of radiology*, vol. 81, no. 9, pp. 2417–2422, 2012.

[170] G. Treglia, M. Salsano, A. Stefanelli, M. V. Mattoli, A. Giordano, and L. Bonomo, "Diagnostic accuracy of 18 F-FDG-PET and PET/CT in patients with Ewing sarcoma family tumours: a systematic review and a meta-analysis," *Skeletal radiology*, vol. 41, no. 3, pp. 249–256, 2012.

[171] H. Adams, T. Kwee, B. de Keizer, R. Fijnheer, J. de Klerk, A. Littooij, and R. Nievelstein, "Systematic review and meta-analysis on the diagnostic performance of FDG-PET/CT in detecting bone marrow involvement in newly diagnosed Hodgkin lymphoma: is bone marrow biopsy still necessary?" *Annals of Oncology*, vol. 25, no. 5, pp. 921–927, 2014.

[172] A. Pfannenberg, P. Aschoff, K. Brechtel, M. Müller, M. Klein, R. Bares, C. Claussen, and S. Eschmann, "Value of contrast-enhanced multiphase CT in combined PET/CT protocols for oncological imaging," *The British journal of radiology*, 2014.

[173] L. S. Mortensen, J. Johansen, J. Kallehauge, H. Primdahl, M. Busk, P. Lassen, J. Alsner, B. S. Sørensen, K. Toustrup, S. Jakobsen *et al.*, "Faza pet/ct hypoxia imaging in patients with squamous cell carcinoma of the head and neck treated with radiotherapy: results from the dahanca 24 trial," *Radiotherapy and Oncology*, vol. 105, no. 1, pp. 14–20, 2012.

[174] B. B. Koolen, F. van der Leij, W. V. Vogel, E. J. T. Rutgers, M.-J. T. Vrancken Peeters, P. H. Elkhuizen, and R. A. Valdés Olmos, "Accuracy of 18F-FDG PET/CT for primary tumor visualization and staging in T1 breast cancer," *Acta Oncologica*, vol. 53, no. 1, pp. 50–57, 2014.

[175] H. Nunes, P. Soler, and D. Valeyre, "Pulmonary sarcoidosis," *Allergy*, vol. 60, no. 5, pp. 565–582, 2005.

[176] J. Kim, W. Cai, S. Eberl, and D. Feng, "Real-time volume rendering visualization of dual-modality pet/ct images with interactive fuzzy thresholding segmentation," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 11, no. 2, pp. 161–169, March 2007.

[177] Y. Jung, J. Kim, S. Eberl, M. Fulham, and D. D. Feng, "Visibility-driven PET-CT visualisation with region of interest (ROI) segmentation," *The Visual Computer*, vol. 29, no. 6-8, pp. 805–815, 2013.

[178] S. H. GmbH. (2015) syngo fusion package. https://www.healthcare.siemens.com/angio/options-and-upgrades/clinical-software-applications/syngo-inspace-3d-3d-fusion. [Online]. Available: https://www.healthcare.siemens.com/angio/options-and-upgrades/clinical-software-applications/syngo-inspace-3d-3d-fusion

[179] G. J. Stanisz, E. E. Odrobina, J. Pun, M. Escaravage, S. J. Graham, M. J. Bronskill, and R. M. Henkelman, "T1, T2 relaxation and magnetization transfer in tissue at 3T," *Magnetic Resonance in Medicine*, vol. 54, no. 3, pp. 507–512, 2005.

[180] G. E. Gold, E. Han, J. Stainsby, G. Wright, J. Brittain, and C. Beaulieu, "Musculoskeletal mri at 3.0 t: relaxation times and image contrast," *American Journal of Roentgenology*, vol. 183, no. 2, pp. 343–351, 2004.

[181] A. Gupta, H. Baradaran, A. D. Schweitzer, H. Kamel, A. Pandya, D. Delgado, A. Dunning, A. I. Mushlin, and P. C. Sanelli, "Carotid Plaque MRI and Stroke Risk A Systematic Review and Meta-analysis," *Stroke*, vol. 44, no. 11, pp. 3071–3077, 2013.

[182] S. Naruse, Y. Horikawa, C. Tanaka, K. Hirakawa, H. Nishikawa, and K. Yoshizaki, "Significance of proton relaxation time measurement in brain edema, cerebral infarction and brain tumors," *Magnetic resonance imaging*, vol. 4, no. 4, pp. 293–304, 1986.

[183] M. E. Shenton, C. C. Dickey, M. Frumin, and R. W. McCarley, "A review of MRI findings in schizophrenia," *Schizophrenia research*, vol. 49, no. 1, pp. 1–52, 2001.

[184] P. Brambilla, A. Hardan, S. U. di Nemi, J. Perez, J. C. Soares, and F. Barale, "Brain anatomy and development in autism: review of structural MRI studies," *Brain research bulletin*, vol. 61, no. 6, pp. 557–569, 2003.

[185] M. A. Haider, T. H. van der Kwast, J. Tanguay, A. J. Evans, A.-T. Hashmi, G. Lockwood, and J. Trachtenberg, "Combined T2-weighted and diffusion-weighted MRI for localization of prostate cancer," *American Journal of Roentgenology*, vol. 189, no. 2, pp. 323–328, 2007.

[186] F. Sardanelli, G. M. Giuseppetti, P. Panizza, M. Bazzocchi, A. Fausto, G. Simonetti, V. Lattanzio, and A. Del Maschio, "Sensitivity of MRI versus mammography for detecting foci of multifocal, multicentric breast cancer in fatty and dense breasts using the whole-breast pathologic examination as a gold standard," *American Journal of Roentgenology*, vol. 183, no. 4, pp. 1149–1157, 2004.

[187] F. M. Gunning-Dixon, A. M. Brickman, J. C. Cheng, and G. S. Alexopoulos, "Aging of cerebral white matter: a review of mri findings," *International journal of geriatric psychiatry*, vol. 24, no. 2, p. 109, 2009.

[188] O. Wieben, C. Francois, and S. B. Reeder, "Cardiac MRI of ischemic heart disease at 3T: potential and challenges," *European journal of radiology*, vol. 65, no. 1, pp. 15–28, 2008.

[189] T. Toyone, K. Takahashi, H. Kitahara, M. Yamagata, M. Murakami, and H. Moriya, "Visualisation of symptomatic nerve roots. prospective study of contrast-enhanced mri in patients with lumbar disc herniation," *Journal of Bone & Joint Surgery, British Volume*, vol. 75, no. 4, pp. 529–533, 1993.

[190] T. Magee and D. Williams, "3.0-T MRI of the supraspinatus tendon," *American Journal of Roentgenology*, vol. 187, no. 4, pp. 881–886, 2006.

[191] T. H. Magee and D. Williams, "Sensitivity and specificity in detection of labral tears with 3.0-T MRI of the shoulder," *American Journal of Roentgenology*, vol. 187, no. 6, pp. 1448–1452, 2006.

[192] N. K. Logothetis, J. Pauls, M. Augath, T. Trinath, and A. Oeltermann, "Neurophysiological investigation of the basis of the fMRI signal," *Nature*, vol. 412, no. 6843, pp. 150–157, 2001.

[193] J. Goense, H. Merkle, and N. K. Logothetis, "High-resolution fMRI reveals laminar differences in neurovascular coupling between positive and negative BOLD responses," *Neuron*, vol. 76, no. 3, pp. 629–639, 2012.

[194] C. B. Kirwan, C. K. Jones, M. I. Miller, and C. E. Stark, "High-resolution fMRI investigation of the medial temporal lobe," *Human brain mapping*, vol. 28, no. 10, pp. 959–966, 2007.

[195] B. Knutson, A. Westdorp, E. Kaiser, and D. Hommer, "FMRI visualization of brain activity during a monetary incentive delay task," *Neuroimage*, vol. 12, no. 1, pp. 20–27, 2000.

[196] G. Winterer, C. M. Adams, D. W. Jones, and B. Knutson, "Volition to actionan event-related fMRI study," *Neuroimage*, vol. 17, no. 2, pp. 851–858, 2002.

[197] L. Rueckert, N. Lange, A. Partiot, I. Appollonio, I. Litvan, D. Le Bihan, and J. Grafman, "Visualizing cortical activation during mental calculation with functional MRI," *Neuroimage*, vol. 3, no. 2, pp. 97–103, 1996.

[198] A. Ishai, J. V. Haxby, and L. G. Ungerleider, "Visual imagery of famous faces: effects of memory and attention revealed by fmri," *Neuroimage*, vol. 17, no. 4, pp. 1729–1741, 2002.

[199] F. Rößler, E. Tejada, T. Fangmeier, T. Ertl, and M. Knauff, "GPU-based Multi-Volume Rendering for the Visualization of Functional Brain Images." in *SimVis*, vol. 2006, 2006, pp. 305–18.

[200] N. Archip, O. Clatz, S. Whalen, D. Kacher, A. Fedorov, A. Kot, N. Chrisochoides, F. Jolesz, A. Golby, P. M. Black *et al.*, "Non-rigid alignment of pre-operative MRI, fMRI, and DT-MRI with intra-operative MRI for enhanced visualization and navigation in image-guided neurosurgery," *Neuroimage*, vol. 35, no. 2, pp. 609–624, 2007.

[201] A. Eklund, O. Friman, M. Andersson, and H. Knutsson, "A GPU accelerated interactive interface for exploratory functional connectivity analysis of fMRI data," in *Image Processing (ICIP), 2011 18th IEEE International Conference On.* IEEE, 2011, pp. 1589–1592.

[202] P. S. Calhoun, B. S. Kuszyk, D. G. Heath, J. C. Carley, and E. K. Fishman, "Three-dimensional Volume Rendering of Spiral CT Data: Theory and Method," *RadioGraphics*, vol. 19, no. 3, pp. 745–764, 1999, pMID: 10336201. [Online]. Available: http://dx.doi.org/10.1148/radiographics.19.3.g99ma14745

[203] S. Napel, M. P. Marks, G. D. Rubin, M. D. Dake, C. H. McDonnell, S. M. Song, D. R. Enzmann, and J. R B Jeffrey, "Ct angiography with spiral ct and maximum intensity projection." *Radiology*, vol. 185, no. 2, pp. 607–610, 1992, pMID: 1410382. [Online]. Available: http://dx.doi.org/10.1148/radiology.185.2.1410382

[204] M. Bhalla, D. P. Naidich, G. McGuinness, J. F. Gruden, B. S. Leitman, and D. I. McCauley, "Diffuse lung disease: assessment with helical CT–preliminary observations of the role of maximum and minimum intensity projection images." *Radiology*, vol. 200, no. 2, pp. 341–347, 1996.

[205] M. Remy-Jardin, J. Remy, B. Gosselin, M. C. Copin, A. Wurtz, and A. Duhamel, "Sliding thin slab, minimum intensity projection technique in the diagnosis of emphysema: histopathologic-CT correlation." *Radiology*, vol. 200, no. 3, pp. 665–671, 1996.

[206] R. Johnson, Thorsten, C. Fink, S. Schöberg, and F. Reiser, Maximilian, *Dual Energy CT in Clinical Practice*. Springer Berlin Heidelberg, 2011.

[207] C. N. De Cecco, V. Buffa, S. Fedeli, M. Luzietti, A. Vallone, R. Ruopoli, V. Miele, M. Rengo, P. Paolantonio, M. M. Enrici *et al.*, "Dual energy ct (dect) of the liver: conventional versus virtual unenhanced images," *European radiology*, vol. 20, no. 12, pp. 2870–2875, 2010.

[208] B. M. Yeh, J. A. Shepherd, Z. J. Wang, H. S. Teh, R. Hartman, and S. Prevrhal, "Dual Energy and Low kVp CT in the abdomen," *AJR. American journal of roentgenology*, vol. 193, no. 1, p. 47, 2009.

[209] J. Ferda, M. Novák, H. Mírka, J. Baxa, E. Ferdová, A. Bednářová, T. Flohr, B. Schmidt, E. Klotz, and B. Kreuzberg, "The assessment of intracranial bleeding with virtual unenhanced imaging by means of dual-energy CT angiography," *European radiology*, vol. 19, no. 10, pp. 2518–2522, 2009.

[210] A. Graser, T. R. Johnson, H. Chandarana, and M. Macari, "Dual energy CT: preliminary observations and potential clinical applications in the abdomen," *European radiology*, vol. 19, no. 1, pp. 13–23, 2009.

[211] A. Graser, T. R. Johnson, E. M. Hecht, C. R. Becker, C. Leidecker, M. Staehler, C. G. Stief, H. Hildebrandt, M. C. Godoy, M. E. Finn *et al.*, "Dual-energy ct in patients suspected of having renal masses: Can virtual nonenhanced images replace true nonenhanced images? 1," *Radiology*, vol. 252, no. 2, pp. 433–440, 2009.

[212] D. P. Clark, K. Ghaghada, E. J. Moding, D. G. Kirsch, and C. T. Badea, "In vivo characterization of tumor vasculature using iodine and gold nanoparticles and dual energy micro-CT," *Physics in medicine and biology*, vol. 58, no. 6, p. 1683, 2013.

[213] A. So, J. Hsieh, S. Narayanan, J.-B. Thibault, Y. Imai, S. Dutta, J. Leipsic, J. Min, T. LaBounty, and T.-Y. Lee, "Dual-energy CT and its potential use for quantitative myocardial CT perfusion," *Journal of cardiovascular computed tomography*, vol. 6, no. 5, pp. 308–317, 2012.

[214] Z. Pan, L. Pang, B. Ding, C. Yan, H. Zhang, L. Du, B. Wang, Q. Song, K. Chen, and F. Yan, "Gastric cancer staging with dual energy spectral CT imaging," *PloS one*, vol. 8, no. 2, p. e53651, 2013.

[215] K. N. Glazebrook, L. S. Guimarães, N. S. Murthy, D. F. Black, T. Bongartz, N. J. Manek, S. Leng, J. G. Fletcher, and C. H. McCollough, "identification of intraarticular and periarticular uric acid crystals with dual-energy ct: initial evaluation," *Radiology*, vol. 261, no. 2, pp. 516–524, 2011.

[216] C. Sun, F. Miao, X.-m. Wang, T. Wang, R. Ma, D.-p. Wang, and C. Liu, "An initial qualitative study of dual-energy CT in the knee ligaments," *Surgical and Radiologic Anatomy*, vol. 30, no. 5, pp. 443–447, 2008.

[217] Y. Watanabe, K. Uotani, T. Nakazawa, M. Higashi, N. Yamada, Y. Hori, S. Kanzaki, T. Fukuda, T. Itoh, and H. Naito, "Dual-energy direct bone removal CT angiography for evaluation of intracranial aneurysm or stenosis: comparison with conventional digital subtraction angiography," *European radiology*, vol. 19, no. 4, pp. 1019–1024, 2009.

[218] D. N. Tran, M. Straka, J. E. Roos, S. Napel, and D. Fleischmann, "Dual-energy CT discrimination of iodine and calcium: experimental results and implications for lower extremity CT angiography," *Academic radiology*, vol. 16, no. 2, pp. 160–171, 2009.

[219] Siemens Healthcare Global. (2015) CT Scanner - SOMATOM Force - Technical Specification - Siemens Healthcare Global. [Online]. Available: http://www.healthcare.siemens.com/computed-tomography/dual-source-ct/somatom-force/technical-specifications

[220] M. P. Shikhaliev and S. G. Fritz, "Photon counting spectral CT versus conventional CT: comparative evaluation for breast imaging application," *Physics in Medicine and Biology*, vol. 56, no. 7, p. 1905, 2011. [Online]. Available: http://stacks.iop.org/0031-9155/56/i=7/a=001

[221] M. P. Shikhaliev, "Soft tissue imaging with photon counting spectroscopic CT," *Physics in Medicine and Biology*, vol. 60, no. 6, p. 2453, 2015. [Online]. Available: http://stacks.iop.org/0031-9155/60/i=6/a=2453

[222] S. Perandini, N. Faccioli, A. Zaccarella, T. Re, and R. P. Mucelli, "The diagnostic contribution of CT volumetric rendering techniques in routine practice," *The Indian Journal of Radiology & Imaging*, vol. 20, no. 2, pp. 92–97, May 2010. [Online]. Available: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2890933/

[223] E. K. Fishman, D. R. Ney, D. G. Heath, F. M. Corl, n. K. M. Horto, and P. T. Johnson, "Volume Rendering versus Maximum Intensity Projection in CT Angiography: What Works Best, When, and Why," *RadioGraphics*, vol. 26, no. 3, pp. 905–922, 2006, pMID: 16702462. [Online]. Available: http://dx.doi.org/10.1148/rg.263055186

[224] D. Pan, E. Roessl, J.-P. Schlomka, S. D. Caruthers, A. Senpan, M. J. Scott, J. S. Allen, H. Zhang, G. Hu, P. J. Gaffney, E. T. Choi, V. Rasche, S. A. Wickline, R. Proksa, and G. M. Lanza, "Computed Tomography in Color: NanoK-Enhanced Spectral CT Molecular Imaging," *Angewandte Chemie*, vol. 122, no. 50, pp. 9829–9833, 2010. [Online]. Available: http://dx.doi.org/10.1002/ange.201005657

[225] D. Pan, C. O. Schirra, A. Senpan, A. H. Schmieder, A. J. Stacy, E. Roessl, A. Thran, S. A. Wickline, R. Proska, and G. M. Lanza, "An early investigation of ytterbium nanocolloids for selective and quantitative multicolor spectral CT imaging," *ACS nano*, vol. 6, no. 4, pp. 3364–3370, 2012.

[226] A. Butler, N. Anderson, R. Tipples, N. Cook, R. Watts, J. Meyer, A. Bell, T. Melzer, and P. Butler, "Bio-medical X-ray imaging with spectroscopic pixel detectors," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 591, no. 1, pp. 141 – 146, 2008, radiation Imaging Detectors 2007 - Proceedings of the 9th International Workshop on Radiation Imaging Detectors.

[227] J. P. Ronaldson, R. Zainon, N. J. A. Scott, S. P. Gieseg, A. P. Butler, P. H. Butler, and N. G. Anderson, "Toward quantifying the composition of soft tissues by spectral CT with Medipix3," *Medical Physics*, vol. 39, no. 11, pp. 6847–6857, 2012. [Online]. Available: http://scitation.aip.org/content/aapm/journal/medphys/39/11/10.1118/1.4760773

[228] W. Cai and G. Sakas, "Data intermixing and multi-volume rendering," *Comput. Graph. Forum*, pp. 359–368, 1999.

[229] N. Pan, H. Liu, N. de Ruiter, and R. Grasset, "Improving the image quality of spectral CT volume rendering," in *Image and Vision Computing New Zealand, 2009. IVCNZ '09. 24th International Conference*, nov. 2009, pp. 203 –208.

[230] University of Brussels. (2015) Virtual animation of the kinematics of the human for industrial, educational and research purposes. [Online]. Available: http://www.ulb.ac.be/project/vakhum/public_dataset/public-data.htm

[231] B. J. Erickson, "Irreversible Compression of Medical Images," *Journal of Digital Imaging*, vol. 15, pp. 5–14, 2002, 10.1007/s10278-002-0001-z. [Online]. Available: http://dx.doi.org/10.1007/s10278-002-0001-z

[232] H. W. Venema, F. J. H. Hulsmans, and G. J. den Heeten, "CT Angiography of the Circle of Willis and Intracranial Internal Carotid Arteries: Maximum Intensity Projection with Matched Mask Bone EliminationFeasibility Study 1," *Radiology*, vol. 218, no. 3, pp. 893–898, 2001.

[233] C. Balazs, L. Mroz, H. Hauser, A. König, and E. Gröller, "Fast Visualization of Object Contours by Non-Photorealistic Volume Rendering," *Computer Graphics Forum*, vol. 20, no. 3, pp. 452–460, 2001. [Online]. Available: http://dx.doi.org/10.1111/1467-8659.00538

[234] M. Hadwiger, P. Ljung, C. R. Salama, and T. Ropinski, "Advanced illumination techniques for GPU-based volume raycasting," in *SIGGRAPH '09: ACM SIGGRAPH 2009 Courses*. New York, NY, USA: ACM, 2009, pp. 1–166.

[235] J. Chen and J. Thropp, "Review of Low Frame Rate Effects on Human Performance," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 37, no. 6, pp. 1063–1076, Nov 2007.

[236] M. Claypool, K. Claypool, and F. Damaa, "The effects of frame rate and resolution on users playing first person shooter games," in *Electronic Imaging 2006*. International Society for Optics and Photonics, 2006, pp. 607 101–607 101.

[237] S. Shimojo and K. Nakayama, "Real world occlusion constraints and binocular rivalry," *Vision research*, vol. 30, no. 1, pp. 69–80, 1990.

[238] D. Escott, E. J. amd Rubinstein, "Free DICOM Image Viewing and Processing Software for Your Desktop Computer: Whats Available and What It Can Do for You," *Radiographics*, vol. 23, pp. 1341–1357, 2003.

[239] Medixant. (2015) DICOM Viewer - RadiAnt. [Online]. Available: http://www.radiantviewer.com/

[240] C. P. Gribble and S. G. Parker, "Enhancing interactive particle visualization with advanced shading models," in *Proceedings of the 3rd symposium on Applied perception in graphics and visualization*. ACM, 2006, pp. 111–118.

[241] L. C. Wanger, J. A. Ferwerda, and D. P. Greenberg, "Perceiving spatial relationships in computer-generated images," *IEEE Computer Graphics and Applications*, no. 3, pp. 44–51, 1992.

[242] D. Kersten, D. C. Knill, P. Mamassian, and I. Bülthoff, "Illusory motion from shadows." 1996.

[243] P. Rheingans and C. Landreth, "Perceptual principles for effective visualizations," in *Perceptual Issues in Visualization*. Springer, 1995, pp. 59–73.

[244] Digia Ltd. (2014) Qt project. http://qt-project.org/.

[245] M. Weiler, R. Westermann, C. Hansen, K. Zimmermann, and T. Ertl, "Level-of-detail volume rendering via 3D textures," in *Proceedings of the 2000 IEEE symposium on Volume visualization*. ACM, 2000, pp. 7–13.

[246] R. Westermann and T. Ertl, "Efficiently using graphics hardware in volume rendering applications," in *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. ACM, 1998, pp. 169–177.

[247] S. Grimm, S. Bruckner, A. Kanitsar, and E. Gröller, "Flexible direct multi-volume rendering in dynamic scenes," in *VMV*, 2004, pp. 379–386.

[248] A. Rosset, L. Spadola, and O. Ratib, "OsiriX: an open-source software for navigating in multidimensional DICOM images." *Journal of Digital Imaging*, vol. 17, no. 0897-1889 (Linking), pp. 205–216, 2004.

[249] U.S. Department of Health & Human Services. (2015) User interface elements. [Online]. Available: http://www.usability.gov/how-to-and-tools/methods/user-interface-elements.html

[250] *DICOM Part 10: Media Storage and File Format for Media Interchange*, National Electrical Manufacturers Association Std., 2014b. [Online]. Available: http://medical.nema.org/medical/dicom/current/output/html/part10.html

[251] IEEE, "IEEE Standard for Floating-Point Arithmetic," *IEEE Std 754-2008*, pp. 1–70, Aug 2008.

[252] OFFIS e. V. (2015) DCMTK - DICOM Toolkit. [Online]. Available: http://dicom.offis.de/dcmtk.php.en

[253] A. Buades, B. Coll, and J.-M. Morel, "The staircasing effect in neighborhood filters and its solution," *Image Processing, IEEE Transactions on*, vol. 15, no. 6, pp. 1499–1505, 2006.

[254] *ISO 12640-2:2004 Graphic technology – Prepress digital data exchange – Part 2: XYZ/sRGB encoded standard colour image data (XYZ/SCID)*, ISO Std.

[255] L. Szirmay-Kalos, B. Tóth, and M. Magdics, "Free path sampling in high resolution inhomogeneous participating media," in *Computer Graphics Forum*, vol. 30, no. 1. Wiley Online Library, 2011, pp. 85–97.

[256] J. Gallier, *Curves and Surfaces in Geometric Modeling: Theory & Algorithms*. Morgan Kaufmann, 1999.

[257] J. Hughes, A. van Dam, M. McGuire, D. Sklar, J. Foley, S. Feiner, and K. Akeley, *Computer Graphics: Principles and Practice (3rd Edition)*. Addison-Wesley Professional, 2013.

[258] J. Beyer, M. Hadwiger, S. Wolfsberger, and K. Buhler, "High-Quality Multimodal Volume Rendering for Preoperative Planning of Neurosurgical Interventions," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 6, pp. 1696–1703, Nov 2007.

[259] S. Lefebre, S. Hornus, and F. Neyret, *GPU Gems 2*. Addison-Wesley, 2005, ch. Octree Textures on the GPU, pp. 595–613.

[260] D. R. Horn, J. Sugerman, M. Houston, and P. Hanrahan, "Interactive k-d tree GPU raytracing," in *Proceedings of the 2007 symposium on Interactive 3D graphics and games*, ser. I3D '07. New York, NY, USA: ACM, 2007, pp. 167–174. [Online]. Available: http://doi.acm.org/10.1145/1230100.1230129

[261] T. Klein, M. Strengert, S. Stegmaier, and T. Ertl, "Exploiting frame-to-frame coherence for accelerating high-quality volume raycasting on graphics hardware," in *Visualization, 2005. VIS 05. IEEE*, oct. 2005, pp. 223 – 230.

[262] P. Ljung, "Adaptive Sampling in Single Pass, GPU-based Raycasting of Multiresolution Volumes," in *Proceedings Eurographics/IEEE International Workshop on Volume Graphics 2006, Boston, USA*, 2006.

[263] W. Li, K. Mueller, and A. Kaufman, "Empty Space Skipping and Occlusion Clipping for Texture-based Volume Rendering," in *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, ser. VIS '03. Washington,

DC, USA: IEEE Computer Society, 2003, pp. 42–. [Online]. Available: http://dx.doi.org/10.1109/VISUAL.2003.1250388

[264] C. Lundström, P. Ljung, and A. Ynnerman, "Extending and simplifying transfer function design in medical volume rendering using local histograms," in *Euro-Graphics/IEEE Symposium on Visualization*, 2005, pp. 263–270.

[265] R. M. Subramaniam, C. Kim, P. Scally, and B. Tress, "Medical student radiology training: what are the objectives for contemporary medical practice?" *Academic radiology*, vol. 10, no. 3, pp. 295–300, 2003.

[266] J. Kniss, G. Kindlmann, and C. Hansen, "Multidimensional transfer functions for interactive volume rendering," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 8, no. 3, pp. 270–285, 2002.

[267] H. Guo, H. Xiao, and X. Yuan, "Multi-dimensional transfer function design based on flexible dimension projection embedded in parallel coordinates," in *Pacific Visualization Symposium (PacificVis), 2011 IEEE*. IEEE, 2011, pp. 19–26.

[268] X. Zhao and A. E. Kaufman, "Multi-dimensional reduction and transfer function design using parallel coordinates." in *Volume Graphics*. Citeseer, 2010, pp. 69–76.

[269] J. Barnes, "Characteristics and control of contrast in CT." *Radiographics*, vol. 12, no. 4, pp. 825–837, 1992.

[270] E. Krupinski, T. Ovitt, and M. Evanoff, "Use of image processing presets in chest radiography," *Journal of digital imaging*, vol. 10, pp. 181–182, 1997.

[271] E. L. Siegel, J. N. Diaconis, S. Pomerantz, R. Allman, and B. Briscoe, "Making filmless radiology work," *Journal of digital imaging*, vol. 8, no. 4, pp. 151–155, 1995.

[272] R. Wolz, C. Chu, K. Misawa, K. Mori, and D. Rueckert, "Multi-organ Abdominal CT Segmentation Using Hierarchically Weighted Subject-Specific Atlases," in *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2012*, ser. Lecture Notes in Computer Science, N. Ayache, H. Delingette, P. Golland, and K. Mori, Eds. Springer Berlin Heidelberg, 2012, vol. 7510, pp. 10–17. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-33415-3_2

[273] S. Hu, E. A. Hoffman, and J. M. Reinhardt, "Automatic lung segmentation for accurate quantitation of volumetric X-ray CT images," *Medical Imaging, IEEE Transactions on*, vol. 20, no. 6, pp. 490–498, 2001.

[274] J. A. Maldjian, J. Chalela, S. E. Kasner, D. Liebeskind, and J. A. Detre, "Automated CT segmentation and analysis for acute middle cerebral artery stroke," *American journal of neuroradiology*, vol. 22, no. 6, pp. 1050–1055, 2001.

[275] C. R. Salama, M. Keller, and P. Kohlmann, "High-level user interfaces for transfer function design with semantics," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 12, no. 5, pp. 1021–1028, 2006.

[276] K. H. Lee, Y. H. Kim, S. Hahn, K. W. Lee, T. J. Kim, S.-B. Kang, and J. H. Shin, "Computed tomography diagnosis of acute appendicitis: advantages of reviewing thin-section datasets using sliding slab average intensity projection technique," *Investigative radiology*, vol. 41, no. 7, pp. 579–585, 2006.

[277] M. Remy-Jardin, J. Remy, D. Artaud, F. Deschildre, and A. Duhamel, "Diffuse infiltrative lung disease: clinical value of sliding-thin-slab maximum intensity projection CT scans in the detection of mild micronodular patterns." *Radiology*, vol. 200, no. 2, pp. 333–339, 1996.

[278] Y. Sato, N. Shiraga, S. Nakajima, S. Tamura, and R. Kikinis, "Local maximum intensity projection (lmip: A new rendering method for vascular visualization," *Journal of computer assisted tomography*, vol. 22, no. 6, pp. 912–917, 1998.

[279] (2004) Elsevier Inc.: Introduction to CT Physics. [Online]. Available: http://www.intl.elsevierhealth.com/e-books/pdf/940.pdf

[280] S. Bruckner, S. Grimm, A. Kanitsar, and M. E. Grller, "Illustrative context-preserving volume rendering," in *IN PROCEEDINGS OF EUROVIS 2005*, 2005, pp. 69–76.

[281] S. K. Feiner and D. D. Seligmann, "Cutaways and ghosting: satisfying visibility constraints in dynamic 3d illustrations," *The Visual Computer*, vol. 8, no. 5-6, pp. 292–302, 1992.

[282] B. Rogers, M. Graham *et al.*, "Motion parallax as an independent cue for depth perception," *Perception*, vol. 8, no. 2, pp. 125–134, 1979.

[283] B. Rogers and M. Graham, "Similarities between motion parallax and stereopsis in human depth perception," *Vision research*, vol. 22, no. 2, pp. 261–270, 1982.

[284] M. C. Stone, K. Fishkin, and E. A. Bier, "The movable filter as a user interface tool," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '94. New York, NY, USA: ACM, 1994, pp. 306–312. [Online]. Available: http://doi.acm.org/10.1145/191666.191774

[285] A. Fuhrmann and E. Gröller, "Real-time Techniques for 3D Flow Visualization," in *Proceedings of the Conference on Visualization '98*, ser. VIS '98. Los Alamitos, CA, USA: IEEE Computer Society Press, 1998, pp. 305–312. [Online]. Available: http://dl.acm.org/citation.cfm?id=288216.288296

[286] J. Looser, R. Grasset, and M. Billinghurst, "A 3D Flexible and Tangible Magic Lens in Augmented Reality," in *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, ser. ISMAR '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 1–4. [Online]. Available: http://dx.doi.org/10.1109/ISMAR.2007.4538825

[287] M. Rohs, J. Schöning, M. Raubal, G. Essl, and A. Krüger, "Map navigation with mobile devices: virtual versus physical movement with and without visual context," in *Proceedings of the 9th international conference on Multimodal interfaces*. ACM, 2007, pp. 146–153.

[288] J. Kruger, J. Schneider, and R. Westermann, "Clearview: An interactive context preserving hotspot visualization technique," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 12, no. 5, pp. 941–948, 2006.

[289] E. Monclús, J. Díaz, I. Navazo, and P.-P. Vázquez, "The virtual magic lantern: an interaction metaphor for enhanced medical data inspection," in *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*. ACM, 2009, pp. 119–122.

[290] D. Kalkofen, E. Mendez, and D. Schmalstieg, "Interactive Focus and Context Visualization for Augmented Reality," in *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, ser. ISMAR '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 1–10. [Online]. Available: http://dx.doi.org/10.1109/ISMAR.2007.4538846

[291] G. Welch, M. Noland, and G. Bishop, "Complementary tracking and two-handed interaction for remote 3D medical consultation with a PDA," in *Proceedings of Trends and Issues in Tracking for Virtual Environments, Workshop at the IEEE Virtual Reality 2007 Conference*. Charlotte, NC USA), Shaker, 2007.

[292] S. F. Thieme, C. R. Becker, M. Hacker, K. Nikolaou, M. F. Reiser, and T. R. Johnson, "Dual energy CT for the assessment of lung perfusioncorrelation to scintigraphy," *European journal of radiology*, vol. 68, no. 3, pp. 369–374, 2008.

[293] R. Huang, K.-L. Ma, P. McCormick, and W. Ward, "Visualizing industrial CT volume data for nondestructive testing applications," in *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*. IEEE Computer Society, 2003, p. 72.

[294] V. Sekhar, S. Neo, L. H. Yu, A. Trigg, and C. C. Kuo, "Non-destructive testing of a high dense small dimension through silicon via (TSV) array structures by using 3D X-ray computed tomography method (CT scan)," in *Electronics Packaging Technology Conference (EPTC), 2010 12th*. IEEE, 2010, pp. 462–466.

[295] J. Plate, T. Holtkaemper, and B. Froehlich, "A flexible multi-volume shader framework for arbitrarily intersecting multi-resolution datasets," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 6, pp. 1584–1591, 2007.

[296] R. Drzymala, R. Mohan, L. Brewster, J. Chu, M. Goitein, W. Harms, and M. Urie, "Dose-volume histograms," *International Journal of Radiation Oncology*, vol. 21, no. 1, pp. 71–78, 1991, three-Dimensional Photon Treatment Planning Report of the Collaborative Working Group on the Evaluation of Treatment Planning for External Photon Beam Radiotherapy. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0360301691901684

[297] B. Fishman and B. Schachter, "Computer display of height fields," *Computers & Graphics*, vol. 5, no. 2, pp. 53 – 60, 1980. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0097849380900072

[298] J. C. Roberts, "Multiple view and multiform visualization," in *Electronic Imaging*. International Society for Optics and Photonics, 2000, pp. 176–185.

[299] S. Seifert, M. Kelm, M. Moeller, S. Mukherjee, A. Cavallaro, M. Huber, and D. Comaniciu, "Semantic annotation of medical images," in *Proc. SPIE*, vol. 7628, 2010, pp. 762 808–762 808–8. [Online]. Available: http://dx.doi.org/10.1117/12.844207

[300] D. Rubin, P. Mongkolwat, V. Kleper, K. Supekar, and D. Channin, "Annotation and Image Markup: Accessing and Interoperating with the Semantic Content in Medical Imaging," *Intelligent Systems, IEEE*, vol. 24, no. 1, pp. 57–65, Jan 2009.

[301] *DICOM Part 3: Information Object Definitions*, National Electrical Manufacturers Association Std., 2014b. [Online]. Available: http://medical.nema.org/medical/dicom/current/output/html/part03.html

[302] R. A. Jarvis, "On the identification of the convex hull of a finite set of points in the plane," *Information Processing Letters*, vol. 2, no. 1, pp. 18–21, 1973.

[303] H. Scherf and R. Tilgner, "A new high-resolution computed tomography (CT) segmentation method for trabecular bone architectural analysis," *American journal of physical anthropology*, vol. 140, no. 1, pp. 39–51, 2009.

[304] K. T. Bae, M. L. Giger, C.-T. Chen, and C. E. Kahn Jr, "Automatic segmentation of liver structure in CT images," *Medical Physics*, vol. 20, no. 1, pp. 71–78, 1993.

[305] S. Taheri, S. Ong, and V. Chong, "Level-set segmentation of brain tumors using a threshold-based speed function," *Image and Vision Computing*, vol. 28, no. 1, pp. 26 – 37, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0262885609000675

[306] C. W. Chen, J. Luo, and K. Parker, "Image segmentation via adaptive k-mean clustering and knowledge-based morphological operations with biomedical applications," *Image Processing, IEEE Transactions on*, vol. 7, no. 12, pp. 1673–1683, Dec 1998.

[307] M. Prastawa, E. Bullitt, S. Ho, and G. Gerig, "A brain tumor segmentation framework based on outlier detection," *Medical Image Analysis*, vol. 8, no. 3, pp. 275 – 283, 2004, medical Image Computing and Computer-Assisted Intervention - {MICCAI} 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1361841504000295

[308] Y. Yang, A. Tannenbaum, and D. Giddens, "Knowledge-based 3d segmentation and reconstruction of coronary arteries using ct images," in *Engineering in Medicine and Biology Society, 2004. IEMBS'04. 26th Annual International Conference of the IEEE*, vol. 1. IEEE, 2004, pp. 1664–1666.

[309] L.-J. Zhang, S.-Y. Wu, J.-B. Niu, Z.-L. Zhang, H. Z. Wang, Y.-E. Zhao, X. Chai, C.-S. Zhou, and G.-M. Lu, "Dual-energy CT angiography in the evaluation of intracranial aneurysms: image quality, radiation dose, and comparison with 3D rotational digital subtraction angiography," *American Journal of Roentgenology*, vol. 194, no. 1, pp. 23–30, 2010.

[310] D. Yoon, K. Lim, C. Choi, B. Cho, S. Oh, and S. Chang, "Detection and characterization of intracranial aneurysms with 16-channel multidetector row CT angiography: a prospective comparison of volume-rendered images and digital subtraction angiography," *American Journal of Neuroradiology*, vol. 28, no. 1, pp. 60–67, 2007.

[311] S. Schroeder, T. Flohr, A. F. Kopp, C. Meisner, A. Kuettner, C. Herdeg, A. Baumbach, and B. Ohnesorge, "Accuracy of density measurements within plaques lo-

cated in artificial coronary arteries by X-ray multislice CT: results of a phantom study," *Journal of computer assisted tomography*, vol. 25, no. 6, pp. 900–906, 2001.

[312] Y. Nakamoto, B. B. Chin, D. L. Kraitchman, L. P. Lawler, L. T. Marshall, and R. L. Wahl, "Effects of Nonionic Intravenous Contrast Agents at PET/CT Imaging: Phantom and Canine Studies," *Radiology*, vol. 227, no. 3, pp. 817–824, 2003, pMID: 12773683. [Online]. Available: http://dx.doi.org/10.1148/radiol.2273020299

[313] M. Devous, J. Lowe, and J. Payne, "Dual-isotope brain SPECT imaging with technetium-99m and iodine-123: validation by phantom studies," *Journal of nuclear medicine*, vol. 33, no. 11, pp. 2030–2035, November 1992. [Online]. Available: http://europepmc.org/abstract/MED/1432168

[314] S. Manglos, F. Thomas, G. Gagne, and B. Hellwig, "Phantom study of breast tissue attenuation in myocardial imaging," *Journal of nuclear medicine*, vol. 34, no. 6, pp. 992–996, June 1993. [Online]. Available: http://europepmc.org/abstract/MED/8509870

[315] H. Kato, M. Kuroda, K. Yoshimura, A. Yoshida, K. Hanamoto, S. Kawasaki, K. Shibuya, and S. Kanazawa, "Composition of MRI phantom equivalent to human tissues," *Medical physics*, vol. 32, no. 10, pp. 3199–3208, October 2005. [Online]. Available: http://dx.doi.org/10.1118/1.2047807

[316] M. C. Yoo, L. Villegas, and D. B. Jones, "Basic ultrasound curriculum for medical students: validation of content and phantom," *Journal of Laparoendoscopic & Advanced Surgical Techniques*, vol. 14, no. 6, pp. 374–379, 2004.

[317] R. Leahy, J. Mosher, M. Spencer, M. Huang, and J. Lewine, "A study of dipole localization accuracy for {MEG} and {EEG} using a human skull phantom," *Electroencephalography and Clinical Neurophysiology*, vol. 107, no. 2, pp. 159–173, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0013469498000571

[318] D. Hay, "Cardiovascular Disease in New Zealand, A Summary of Recent Statistical Information," New Zealand Heart Foundation, Tech. Rep., .2004.

[319] E. Falk, H. Sillesen, P. Muntendam, and V. Fuster, "The High-Risk Plaque Initiative: Primary Prevention of Atherothrombotic Events in the Asymptomatic Population," *Current Atherosclerosis Reports*, vol. 13, no. 5, pp. 359–366, 2011. [Online]. Available: http://dx.doi.org/10.1007/s11883-011-0193-0

[320] R. Ross, "The pathogenesis of atherosclerosis: a perspective for the 1990s," *Nature*, vol. 362, no. 6423, pp. 801–809, Apr. 1993. [Online]. Available: http://dx.doi.org/10.1038/362801a0

[321] M. Naghavi, P. Libby, E. Falk, S. W. Casscells, S. Litovsky, J. Rumberger, J. J. Badimon, C. Stefanadis, P. Moreno, G. Pasterkamp *et al.*, "From vulnerable plaque to vulnerable patient a call for new definitions and risk assessment strategies: part I," *Circulation*, vol. 108, no. 14, pp. 1664–1672, 2003.

[322] E. Falk, "Why do plaques rupture?" *Circulation*, vol. 86, no. 6 Suppl, pp. III30–42, 1992.

[323] R. Eston, M. Hawes, A. Martin, and T. Reilly, "Human body composition," *Kinanthropometry and exercise physiology laboratory manual*, p. 1, 2009.

[324] P. Frisullo, J. Laverse, R. Marino, and M. Del Nobile, "X-ray computed tomography to study processed meat microstructure," *Journal of Food Engineering*, vol. 94, no. 3, pp. 283–289, 2009.

[325] M. Young, G. Simm, and C. Glasbey, "Computerised tomography for carcass analysis," in *Proceedings of the British Society of Animal Science*, vol. 62, 2001, pp. 255–263.

[326] D. T. Felson, "Epidemiology of hip and knee osteoarthritis." *Epidemiologic reviews*, vol. 10, pp. 1–28, 1987.

[327] New Zealand Ministry of Health. (2015) Arthritis — Ministry of Health NZ. http://www.health.govt.nz/your-health/conditions-and-treatments/diseases-and-illnesses/arthritis. [Online]. Available: http://www.health.govt.nz/your-health/conditions-and-treatments/diseases-and-illnesses/arthritis

[328] Centers for Disease Control and Prevention, "Prevalence of disabilities and associated health conditions among adults - United States, 1999." *MMWR. Morbidity and mortality weekly report*, vol. 50, no. 7, p. 120, 2001.

[329] D. Cumberland, "Hexabrix - a new contrast medium in angiocardiography." *British heart journal*, vol. 45, no. 6, pp. 698–702, 1981.

[330] A. W. Palmer, R. E. Guldberg, and M. E. Levenston, "Analysis of cartilage matrix fixed charge density and three-dimensional morphology via contrast-enhanced microcomputed tomography," *Proceedings of the National Academy of Sciences*, vol. 103, no. 51, pp. 19 255–19 260, 2006. [Online]. Available: http://www.pnas.org/content/103/51/19255.abstract

[331] R. A. Brooks and G. Di Chiro, "Beam hardening in x-ray reconstructive tomography," *Physics in medicine and biology*, vol. 21, no. 3, p. 390, 1976.

[332] L. Yu, S. Leng, and C. H. McCollough, "Dual-Energy CT-Based Monochromatic Imaging," *American Journal of Roentgenology*, vol. 199, pp. S9–S15, 2012. [Online]. Available: http://dx.doi.org/10.2214/AJR.12.9121

[333] A. Coleman and M. Sinclair, "A beam-hardening correction using dual-energy computed tomography," *Physics in medicine and biology*, vol. 30, no. 11, p. 1251, 1985.

[334] T. Devine and J. Wulff, "Cast vs. wrought cobalt-chromium surgical implant alloys," *Journal of biomedical materials research*, vol. 9, no. 2, pp. 151–167, 1975.

[335] D. Sarker and P. Workman, "Pharmacodynamic biomarkers for molecular cancer therapeutics," in *Genomics in Cancer Drug Discovery and Development*, ser. Advances in Cancer Research, G. F. V. W. Garret M. Hampton, Karol Sikora and G. Klein, Eds. Academic Press, 2006, vol. 96, pp. 213 – 268. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0065230X06960084

[336] V. M. Spitzer and D. G. Whitlock, *Atlas of the Visible Human male: reverse engineering of the human body.* DIANE Publishing Company, 2002.

[337] B. A. Garner and M. G. Pandy, "Musculoskeletal model of the upper limb based on the visible human male dataset," *Computer methods in biomechanics and biomedical engineering*, vol. 4, no. 2, pp. 93–126, 2001.

[338] M. J. Ackerman, "The Visible Human Project: a resource for education." *Academic Medicine*, vol. 74, no. 6, pp. 667–70, 1999.

[339] W. DeJarnette, C. Stockham, and S. Wineke, "CT Workflow in a PACS Environment ," DeJarnette Research Systems, Inc. , Tech. Rep., 2005.

[340] S. Guthe, M. Wand, J. Gonser, and W. Straßer, "Interactive rendering of large volume data sets," in *Visualization, 2002. VIS 2002. IEEE.* IEEE, 2002, pp. 53–60.

[341] Sensable Technologies. (2014, August) PHANTOM OMNI. [Online]. Available: http://www.dentsable.com/haptic-phantom-omni.htm

[342] L. Gallo, A. Placitelli, and M. Ciampi, "Controller-free exploration of medical image data: Experiencing the Kinect," in *Computer-Based Medical Systems (CBMS), 2011 24th International Symposium on*, June 2011, pp. 1–6.

[343] zSpace, Inc. (2015) zspace — real world virtual reality. [Online]. Available: http://zspace.com/

Appendix A

Appendix A: Glossary of terms

CT - *Computed Tomography*, a medical, scientific and industrial imaging technology that involves scanning a patient or an object using x-rays and reconstructing projection images to form a 3D volumetric dataset. The dataset can be visualised to reveal internal structure. See section 2.1 for further information.

CUDA - *Compute Unified Device Architecture* - a GPGPU programming language designed by NVIDIA for consumer (GeForce series) and professional-grade (Quadro and Tesla series) graphics cards. See section 2.3 for further information.

DICOM - *Digital Imaging and Communications in Medicine* - a widely-used standard for storing and exchanging medical images, along with the associated metadata (for example, patient information and scan parameters). **Energy volume** - see **Spectral CT dataset**.

GPGPU - *General Purpose Computing on Graphics Processing Units* - using a GPU as a co-processor to perform general purpose work (that is, computation that is not related to the standard graphics rendering pipeline). Easily parallelisable tasks such as matrix multiplication, fast Fourier transforms, or n-body simulation are good examples of problems where GPGPU can be used effectively. See section 2.3 for further information.

GPU - *Graphics Processing Unit*, a hardware unit that is specially designed for massively parallel processing of vector and scalar data for the purpose of rendering 2D or 3D scenes. CPU (Central Processing Unit) and GPU designs are based on different principles: GPUs contain large numbers of small, relatively slow execution units (cores), and emphasise massively parallel computation, while CPUs contain few cores, but achieve excellent single-threaded performance due to ad-

vanced instruction pipelines, branch prediction hardware and multiple levels of fast data caches. See section 2.3 for further information.

Intermixing (in the context of spectral CT imaging) - the process of combining, or fusing data from multiple energy and/or material volumes in order to display relevant information about the structure or composition of the object being studied. Intermixing is a complicated interactive process that has been studied before, but remains a problem for users of the MARS molecular imaging system. A significant part of this thesis (specifically, Chapters 5 and 6), is concerned with the design of intermixing algorithms and graphical user interfaces to facilitate data fusion.

Material decomposition (in the context of spectral CT imaging) - the process of extracting the information about the materials comprising the scanned object. This may be done by processing energy volumes, or by directly reconstructing material volumes from projection images. The end result is a set of volumes that represent the concentrations of a number of materials inside the object. Currently, post-processing with material decomposition algorithms is a standard component of the MARS data processing toolchain (section 3.2).

MARS - *Medipix All-Resolution System* - a collaboration between the University of Canterbury, University of Otago, and other partners worldwide. The aim of the MARS project is to create a complete spectral CT system and introduce it into clinical use. See section 2.1.3 for further information.

Tomographic reconstruction - the process of transforming multiple 2D projections of an object into a single 3D dataset. In the case of CT, a detector rotates around an object or a patient and acquires a series of projection images from different angles. These images are subsequently reconstructed using techniques such as filtered back projection and iterative algebraic reconstruction. See section 2.1 for further information.

Slice (as applied to volumetric datasets) - a single section through a volumetric dataset. In medical imaging, volumetric datasets are generally stored as a series of slices, usually in the DICOM file format.

Spectral CT - *Spectral Computed Tomography* - an emerging medical imaging technology that measures the attenuation of x-rays over multiple (three or more) different energy ranges. Spectral CT is currently being used for pre-clinical research, while human-scale scanners are under development. See section 2.1 for further information.

Spectral CT data types - two different types of datasets that are currently produced by the MARS spectral CT imaging system. The difference lies in the type of post-processing that a spectral CT dataset undergoes after reconstruction. Energy volumes are standard reconstructed CT datasets, as they represent the attenuation of x-rays over a particular energy range. Material volumes are created by processing energy volumes using material decomposition algorithms. The resulting volumetric datasets represent the concentrations of certain materials inside the scanned object. See section 3.2 for further details.

Volumetric dataset (also known as a *volume dataset* or simply as a *volume*) - a dataset with its elements (voxels) positioned on a regular grid in 3D space.

Volume rendering - a family of techniques for visualising volumetric datasets. Volume rendering is often based on ray casting: sending thousands of rays to sample inside the dataset and accumulate colour and opacity. This process gradually builds up an image, which is then displayed to the user. Refer to section 2.2.2.1 for further information.

Voxel - *volumetric pixel* - an element of a volumetric dataset that represents a single value on a regular grid in 3D space. In comparison, a *pixel* (picture element), is a single element on a regular grid in 2D space (usually a 2D image).

Appendix B

Appendix B: XML format for storing local presets in MARS Vision.

This appendix shows an example of the XML format used for storing local presets in MARS Vision. This is a preset created for the Meat1127 dataset (section 9.3). In total, there are 3 transfer functions, but only one (the first, referred to as "TF0") is shown here for brevity.

Listing B.1: Test

```
<Presets>
    <Camera>
        <Film>
            <Width Value="700" />
            <Height Value="500" />
            <Exposure Value="0.83" />
        </Film>
        <Aperture>
            <Size Value="0" />
        </Aperture>
        <Projection>
            <FieldOfView Value="35" />
        </Projection>
        <Focus>
            <FocalDistance Value="0.75" />
        </Focus>
        <From X="-0.133876" Y="0.542082" Z="0.685337" />
        <Target X="0.5" Y="0.5" Z="0.144495" />
        <Up X="0.649409" Y="0.00946334" Z="0.760381" />
        <Pan X="0.00370055" Y="-0.0481934" />
        <MinAaBbBox X="0" Y="0" Z="0.0949" />
        <MaxAaBbBox X="0.9999" Y="0.9242" Z="0.2242" />
        <MinClipBox X="0" Y="0" Z="0" />
        <MaxClipBox X="0" Y="0" Z="0" />
```

```
25        <ClippingBoxVisible Value="0" />
26    </Camera>
27    <TF0>
28        <Nodes>
29            <Node>
30                <NormalizedIntensity Value="0" />
31                <Opacity Value="0" />
32                <Diffuse G="0" R="0" B="0" />
33                <Specular G="0" R="0" B="0" />
34                <Roughness Value="0" />
35            </Node>
36            <Node>
37                <NormalizedIntensity Value="0.07715" />
38                <Opacity Value="0" />
39                <Diffuse G="255" R="255" B="255" />
40                <Specular G="0" R="0" B="0" />
41                <Roughness Value="0" />
42            </Node>
43            <Node>
44                <NormalizedIntensity Value="0.170138" />
45                <Opacity Value="1" />
46                <Diffuse G="51" R="255" B="221" />
47                <Specular G="0" R="0" B="0" />
48            <Roughness Value="0" />
49            </Node>
50            <Node>
51                <NormalizedIntensity Value="1" />
52                <Opacity Value="1" />
53                <Diffuse G="51" R="255" B="221" />
54                <Specular G="0" R="0" B="0" />
55                <Roughness Value="0" />
56            </Node>
57        </Nodes>
58        <DensityScale Value="100" />
59        <ShadingType Value="2" />
60        <GradientFactor Value="10" />
61        <VolumeOpacity Value="1" />
62        <OverlayState Value="0" />
63        <VisibilityState Value="1" />
64        <VolumeName Value="Calcium" />
65    </TF0>
66    <Lighting>
```

```
67      <Lights>
68          <Light Name="Default" Description="">
69              <Theta Value="-78.436" />
70              <Phi Value="19.609" />
71              <Distance Value="3.25" />
72              <Width Value="0.7" />
73              <Height Value="0.69999" />
74              <LockSize Value="1" />
75              <Color G="255" R="255" B="255" />
76              <Intensity Value="50" />
77          </Light>
78      </Lights>
79      <Background>
80          <Enable Value="0" />
81          <TopColor G="170" R="85" B="255" />
82          <MiddleColor G="255" R="255" B="255" />
83          <BottomColor G="31" R="63" B="0" />
84          <Intensity Value="2" />
85      </Background>
86  </Lighting>
87  <ClippingPlane Direction="1" PointX="0" PointY="0.5" PointZ="0"
        DegreesX="0" Visible="0" DegreesY="0" />
88  <Annotations>
89      <Annotation Name="Annotation 1" Green="45" Red="140" Blue="
            255" Visible="1" Type="Points">
90          <Point X="201" Y="125" Z="87" />
91          <Point X="225" Y="151" Z="91" />
92          <Point X="196" Y="146" Z="86" />
93      </Annotation>
94      <Annotation Name="Annotation 2" Green="0" Red="255" Blue="0"
            Visible="1" Type="Points">
95          <Point X="266" Y="242" Z="84" />
96          <Point X="216" Y="214" Z="84" />
97          <Point X="161" Y="210" Z="72" />
98          <Point X="88" Y="239" Z="88" />
99          <Point X="94" Y="161" Z="78" />
100     </Annotation>
101 </Annotations>
102 <MagicLens ContextPreserveMultiplier="4" State="0"
        BorderEnabled="1" InclusionMode="0" BorderWidth="1"
        PreserveContext="0">
103     <Size2D X="0.15" Y="0.15" />
```

```
104         <Pos2D X="0.5" Y="0.5" />
105         <Size3D X="100" Y="100" Z="100" />
106         <Pos3D X="0" Y="0" Z="0" />
107     </MagicLens>
108     <SliceViewSettings>
109         <Saggital SliceIndex="218" CentreX="64.71235205272579"
                CentreY="219.2130925786086" ZoomLevel="1.23624"
                VolumeIndex="0" Level="0.53684" SpectralMode="0" Window="
                1.07367" />
110         <Coronal SliceIndex="218" CentreX="218.6050530366506"
                CentreY="64.18874133486141" ZoomLevel="1.65138"
                VolumeIndex="0" Level="0.82235" SpectralMode="0" Window="
                1.6447" />
111         <Axial SliceIndex="63" CentreX="219.2130925786086" CentreY="
                219.2130925786086" ZoomLevel="1.23624" VolumeIndex="0"
                Level="0.47493" SpectralMode="0" Window="0.94985" />
112         <Arbitrary SliceIndex="0" CentreX="0" CentreY="0" ZoomLevel=
                "0.894531" VolumeIndex="0" Level="0" SpectralMode="0"
                Window="0.0001" />
113     </SliceViewSettings>
114     <Measurements>
115         <Saggital>
116             <Ellipses />
117             <Polygons />
118             <Lines>
119                 <Line_0 Selected="1" Visible="1">
120                     <Start X="-41" Y="149" />
121                     <End X="179" Y="154" />
122                 </Line_0>
123             </Lines>
124         </Saggital>
125         <Coronal>
126         <Ellipses>
127             <Ellipse_0 CentreX="211" CentreY="40" Selected="1"
                    RadiusX="40" RadiusY="38" Visible="1" />
128         </Ellipses>
129         <Polygons />
130         <Lines />
131         </Coronal>
132         <Axial>
133             <Ellipses />
134             <Polygons />
```

```
135             <Lines/>
136         </Axial>
137     </Measurements>
138 </Presets>
```